

≡ Contents

Remembering the Improbable Life of Ed Fredkin (1934–2023) and His World of Ideas and Stories

August 22, 2023

Programmer of the Universe

“OK, so let me tell you...” And so it would begin. A long and colorful story. An elaborate description of a wild idea. In the forty years I knew [Ed Fredkin](#) I heard countless wild ideas and colorful stories from him. He always radiated a certain adventurous joy—together with supreme, almost-childlike confidence. Ed was someone who wanted to independently figure things out for himself, and delighted in presenting his often somewhat-outlandish conclusions—whether about technology, science, business or the world—with dramatic showman-like panache.



In all the years I knew Ed, I’m not sure he ever really listened to anything I said (though he did use [tools I built](#)). He used to like to tell people I’d learned a lot from him. And indeed we had intellectual interests that should have overlapped. But in actuality our ways of thinking about them mostly didn’t connect much at all. But at a personal and social level it was still always a lot of fun being around Ed and being exposed to his unique intense opportunistic energy—with its repeating themes but ever-changing directions.

And there was one way in which Ed and I were very much aligned: both of our lives were deeply influenced by computers and computing. Ed had started with computers in 1956—as part of one of the very first cohorts of programmers. And perhaps on the basis of that experience, he would still, even at the end of his life, matter-of-factly refer to himself as “the

world’s best programmer”. Indeed, so confident was he of his programming prowess that he became convinced that he should in effect be able to write a program for the universe—and make all of physics into a programming problem. It didn’t help that his knowledge of physics was at best spotty (and, for example, I don’t think he ever really learned calculus). But his almost lifelong desire to “program physics” did successfully lead him to the concept of reversible logic, and to what’s now called the “[Fredkin gate](#)”. But it also led him to the idea that the universe must be a giant cellular automaton—whose program he could invent.

I first met Ed in 1982—on an island in the Caribbean he had bought with money from taking public a tech company he’d founded. The year before [I had started studying cellular automata](#), but, unlike Ed, I wasn’t trying to “program” them—to be the universe or anything else. Instead, I was mostly doing what amounted to empirical science, running computer experiments to see what they did, and treating them as part of a computational universe of possible programs “out there to explore”. It wasn’t a methodology I think Ed ever really understood—or cared about. He was a programmer (and inventor), not an empirical scientist. And he was convinced—like a modern analog of an ancient Greek philosopher—that by pure thought he could come up with the whole “clockwork” of the universe.

Central to his picture was the idea that at the bottom of everything was a cellular automaton, with its grid of cells somehow laid out in space. I told Ed countless times that what was known from twentieth-century physics [implied this really couldn’t be](#) how things worked at a fundamental level. I tried to interest Ed in my way of using cellular automata. But Ed wasn’t interested. He was going for what he saw as the big prize: using them to “construct the universe”.

Every few years Ed would tell me he’d made progress—and rather dramatically say things like that he’d “found the electron”. I’d politely ask for details. Then start pointing out that it couldn’t work that way. But soon Ed would be telling a story or talking about some completely different idea—about technology, business or something else.

By the mid-1980s I’d discovered a lot about cellular automata. And I always [felt a bit embarrassed](#) by Ed’s attempt to use them in what seemed to me like a very naive way for fundamental physics—and I worried (as did happen a few times) that people would dismiss my efforts by identifying them with his.

My own career had begun in the 1970s with traditional fundamental physics. And while I didn’t think cellular automata as such could be directly applied to fundamental physics, I

did think that the core computational phenomena I’d discovered through studying cellular automata might be very relevant. And then in the early 1990s I had an idea. In a cellular automaton, space has a fixed grid-like structure. But what if the structure of space is in fact dynamic, and everything in the universe emerges just from the dynamics of that structure? Finally I felt as if there might be a plausible computational foundation for fundamental physics.

I wrote about this in [one chapter](#) of my 2002 book *A New Kind of Science*. I don’t know if Ed ever read what I wrote, but in any case it didn’t seem to affect his idea that the universe was a cellular automaton—and to confuse things further, he told quite a few people that was what I was saying too. At first I found this frustrating—and upsetting—but eventually I realized it was just “Ed being Ed”, and there were still plenty of things to like about Ed.

Nearly twenty years passed. I would see Ed with some regularity. And sometimes I would mention physics. But Ed would just keep talking about his idea that the universe is a cellular automaton. And when we finally made the breakthrough that led in 2020 to our [Physics Project](#) it made me a little sad that I didn’t even try to explain it to Ed. The universe isn’t a cellular automaton. But it is computational. And I think that knowing this would have brought a certain intellectual closure to Ed’s long journey and aspirations around physics.

Ed might have considered physics his single most important quest. But Ed’s life as a whole was filled with a remarkably rich assortment of activities and interests. Computers. Inventions. Companies. Airplanes. MIT. His island. The Soviet Union. Not to mention people, like [Marvin Minsky](#), [John McCarthy](#) and [Richard Feynman](#) (as well as [Tom Watson](#), [Richard Branson](#), and many more). And he would tell stories about all these people and things, and more. Sometimes (particularly later in his life) the stories would repeat. But with remarkable regularity Ed would surprise me with yet another—often at first hard-to-believe—story about a situation or topic that I had no idea he’d ever been involved in.

But what was the “whole Ed story”? I knew a lot of fragments, often quite colorful. But they didn’t seem to fit together into the narrative of a life. And now that Ed is sadly no longer with us, I decided I should really try to “understand Ed” and his story. A few times over the years I had made efforts to ask Ed for systematic historical accounts—and in 2014 I even recorded many hours of oral history with him. But there was clearly much more. And in writing this piece I found myself going through lots of documents and archives—and having

quite a few conversations— and unearthing even yet more stories than I already knew. And in the end there’s a lot to say—and indeed this has turned into the most difficult and complicated [biographical piece](#) I’ve ever written. But I hope that everything I’ve assembled will help tell the often so-wild-you-can’t-make-this-stuff-up story of that most singular individual who I knew all those years.

The Beginning of the Story

Ed never said much to me about his early life. And in fact I think it was only in writing this piece that I even learned he’d grown up in Los Angeles (specifically, East Hollywood). His parents were both (Jewish) Russian immigrants (his father was born in St. Petersburg; his [mother in Odessa](#); they met in LA). His [father](#)’s university engineering studies had been cut short by the Russian Revolution, and he now had a one-man wholesale electronic parts business. His [mother](#) had in her youth been trained as a concert pianist, and died when Ed was 11, leaving a somewhat fragmented family situation. Ed had a [half-sister](#), 14 years older than him, a [brother](#) 6 years older, and a [sister](#) a year older. As he told it in later oral histories, he got interested in both machines and money very early, repairing appliances for a fee even as a tween, and soon learning about the idea of owning stock in companies.

But Ed Fredkin’s first piece of public visibility seems to have come in 1948, when he was 13 years old—and it reminds me so much of many of Ed’s later “self-imposed” adventures. There was at that time an exhibition of historic US documents traveling around the country on a train named the [Freedom Train](#). And when the train came to Los Angeles, the young Ed Fredkin [decided he had to be the first person to see it](#):

He's a Hero Now — the First Boy to Get Aboard the Freedom Train



Story on Page 2, Part 1

NEW MEANING—Thirteen-year-old Edward Fredkin, center, who got up at 2 a.m., braved cold wind and suspicious police to become the first schoolboy here aboard the Freedom Train, leads his junior high school class in the pledge of allegiance.

The *Los Angeles Times* published his account of his adventure—a younger but “quintessentially Ed” story:

'I SAW OUR HISTORY IN MAKING,' BOY SAYS

Proud Young Angeleno Tells His Own Story of Eventful Hours Visiting Freedom Train

Motherless, 13-year-old Edward Fredkin, 2423 Ronda Vista Drive, arose at 2 a.m. Monday and rode his bicycle to the Freedom Train to view America's historic documents. Police picked him up, thinking that he was a runaway, but when they learned of his seriousness, he became the first Los Angeles schoolboy to enter the train.

Edward, a student in Grade A-8 at the Thomas Starr King Junior High School, herewith tells his own story of impressions he gained on that day—first day of the train's visit in Los Angeles.

BY EDWARD FREDKIN

"What I Learned on the Freedom Train"

"When my alarm woke me up at 2 a.m., I was scared, but soon I thought it would be more than worth eight hours of doing nothing to see the Freedom Train.

"The wind blew me down to Exposition Blvd. in quite a hurry and I was alone. By trying to set up some chairs which the wind had blown over, I was quickly picked up by the police, who thought I was a runaway.

"After they found out other-

wise they deposited me in a tent with a heater to keep me warm.

"Time flew as the moon went down and the sun came up—and I was on the Freedom Train.

"Before me was every document I had read and studied about. I saw our history in the making and I saw Washington and Jefferson putting away the Declaration of Independence for me to see today. I saw the surrender papers of World War II that someone else will see tomorrow.

Gets New Impression

"It struck me as a great surprise to actually see what I had so often thought about and left me with a new impression of our country and the fine men who made the U.S.A. the country that it is."

It took Edward just 15 minutes to put his impressions on paper as the rest of his class went about their studies under guidance of Miss Louise Taves, the teacher.

Miss Taves was also proud of the fact that a total of seven students in her class had already visited the famed train.

Ed's record in high school was at best spotty. But as he tells it, he figured out very early a system for improving the odds in multiple-choice tests, and for example in 9th grade got a top score on a newly instituted (multiple-choice) California-wide IQ test. At the end of high school, Ed applied to Caltech (which was only 13 miles away from where he lived), and largely on the basis of his test scores, was admitted. He ended up spending time working various jobs to support himself, didn't do much homework, and by his sophomore year—before having to pick a major—dropped out. In 2015 Ed told me a nice story about his time at Caltech:

In 1952–53, I was a student in Linus Pauling’s class where he lectured Freshman Chemistry at Caltech. After class, one day, I asked Pauling “What is a superconductor at the highest known temperature?” Pauling immediately replied “Niobium Nitride, 18 Kelvin”. I was puzzled because I had never heard of Niobium, so I looked it up and, with some difficulty found a reference that defined it as a European name for the metal Columbium.

Later that same day, reading a Pasadena newspaper, I saw an article about Pauling: It announced that Pauling had just returned from Europe (London is what I recall) where Pauling, as Chairman of the International Committee on the naming of the elements, had decided that henceforth the metal Columbium would be renamed Niobium.

I recently looked into that matter and discovered that evidently that renaming was part of a USA–Europe Compromise... In Europe it had been Wolfram and Niobium, in the USA it had been Tungsten and Columbium.

Europe got its way re Niobium and the USA got its way re Tungsten... Perhaps it was a flip of a coin? Someone might know.

As a Wolfram, I thought you might be interested (and, of course, perhaps all this is old hat to you...).

(For what it’s worth, I actually didn’t know this “Wolfram story”, though the details weren’t quite as dramatic as Ed said: the “[niobium](#)” [decision](#) was actually made in 1949, without Pauling specifically involved, though Pauling [did indeed travel to London](#) just before the beginning of the 1952 school year.)

With his interest in machinery, Ed had always been keen on cars, and in his freshman year at Caltech, he also decided to learn to fly a plane. Ed’s older brother, Norman, had joined the Air Force five years earlier. And when he left Caltech—in 1954 at age 19—Ed joined the Air Force too. (If he hadn’t done that, he would have been drafted into the Army.) Ed’s brother Norman (who would spend his whole career in aviation) [had been involved in the Korean War](#), particularly doing aerial reconnaissance—here pictured with his plane (and, no, there don’t seem to be any Air Force pictures of Ed himself):



By the time Ed joined the Air Force, the Korean War was over. Ed was assigned to an [airbase in Arizona](#), and by the summer of 1955 he had qualified as a fighter pilot. Ed was never officially a “test pilot”, but he told me stories about figuring out how to take his plane higher than anyone else—and achieving weightlessness by flying his plane in a perfect free-fall trajectory by maintaining an eraser floating in midair in front of him.

By 1956 Ed had been grounded from flying as a result of asthma, and was now at an [airbase in Florida](#) as an “intercept controller”—essentially an air traffic controller responsible for guiding fighters to intercept bombers. It was a time when the Air Force was developing the [SAGE \(Semi-Automatic Ground Environment\) air defense system](#)—a huge project whose concept was to use computers to coordinate data from many radars so as to be able to intercept Soviet bombers that might attack the US (cf. [Dr. Strangelove](#), etc.). The center of SAGE development was [Lincoln Lab](#) (then part of MIT) in Lexington, MA—with IBM providing computers, Bell (AT&T) providing telecommunications, [RAND](#) providing algorithms, etc. And in mid-1956 the Air Force sent a group—including Ed—to test the next phase of SAGE. But as Ed tells it, they were soon informed that actually there would be a one-year delay.

At the time, the SAGE project was busily trying to train people about computers, and some people from the Air Force stayed in the Boston area to participate in this. As Ed tells it, however, he was the only one who didn’t drop out of the training—and over the course of a year it taught him “much of what was then known about computer programming and computer hardware design”. There were at the time only a few hundred people in the world

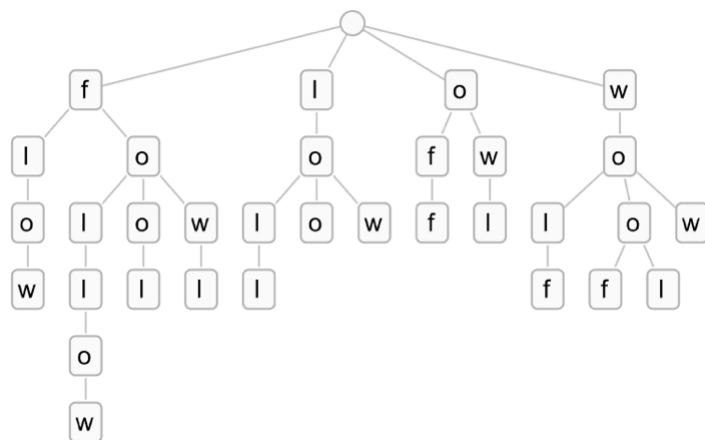
who could call themselves programmers. And Ed was now one of them. (Perhaps he was even “the world’s best”.)

Computers!

Having learned to program, Ed remained at Lincoln Lab, paid by the Air Force, doing what amounted to computational “odd jobs”. Often this had to do with connecting systems together, or coming up with “clever hacks” to overcome particular system limitations. Occasionally it was a little more algorithmic—like when Sputnik was launched in 1957, and Ed got pulled into a piece of “emergency programming” for orbit calculations.

Ed told many stories about “hacking” the bureaucracy at the Air Force (being given a “Secret” stamp so he could read his own documents; avoiding being sent for a year to the Canadian Arctic by finding a loophole associated with his wife being pregnant, etc.)—and in 1958 he left the Air Force (though he would remain a captain in the reserves for many years), but stayed on at Lincoln Lab. Officially he was there as an “administrative assistant”, because—without a degree—that was all they could offer him. But by then he was becoming known as a “computer person”—with lots of ideas. He wanted to start his own company. And (as he tells it) the very first potential customer he visited was an MIT-spinoff acoustics firm called [Bolt Beranek & Newman \(BBN\)](#). And the person he saw there was their “vice president of engineering psychology”—a certain [J. C. R. “Lick” Licklider](#)—who persuaded Ed to join BBN to “teach them about computers”.

It didn’t really come to light until he was at BBN, but while at Lincoln Lab Ed had made what would eventually become his first lasting contribution to computer science. He thought of it as a new way of storing textual information in a computer, and he called it “TRIE memory” (after “reTRIEval”). Nowadays we’d call it the [trie \(or prefix tree\) data structure](#). Here it is for some common words in English made from the letters of “wolf”:



Licklider persuaded Ed to write a paper about tries—which appeared in 1960, and for a couple of decades was essentially Ed’s only academic-style publication:

Trie Memory*

EDWARD FREDKIN, *Bolt Beranek and Newman, Inc., Cambridge, Mass.*

Introduction

Trie memory is a way of storing and retrieving information.¹ It is applicable to information that consists of function-argument (or item-term) pairs—information conventionally stored in unordered lists, ordered lists, or

is simply the binary variable of which the admissible values are *member* and *nonmember*.

At the outset, before a storage is begun, the trie is merely a collection of *registers*. Except for two special registers, which we may call α and δ , every register has a *cell* for each member (type) of the ensemble of alphabetic

The paper has a pretty clear description of tries, even with some nice diagrams:

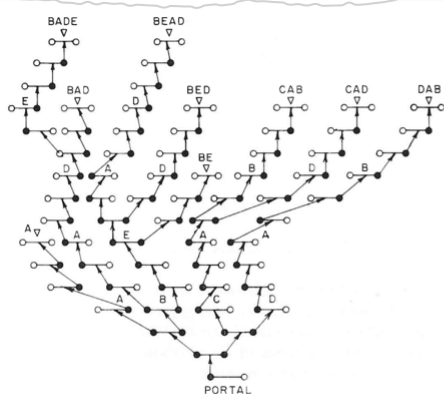


FIG. 5. Storage of words in a character-coded binary tree

one dimension at a time. Then, if the present register has the location $i, j, k, \dots$, its neighbors are the registers with locations $i+1, j, k, \dots; i, j+1, k, \dots; i, j, k+1, \dots$; etc. This is illustrated for a three-dimensional structure in Fig. 6. The number of neighbors is equal to the dimensionality of the structure, and the number of bits required to specify a relative location is only $\log_2 N$.

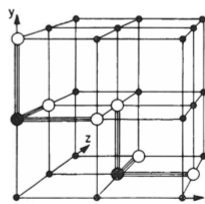


FIG. 6. Three-dimensional memory structure

Even in analyzing the performance of tries, there was only the faintest hint of math in the paper—though Ed realized (probably with input from Licklider) that the efficiency of tries would depend on the [Shannon-style redundancy](#) of what they were storing, and he ran Monte Carlo simulations to investigate this:

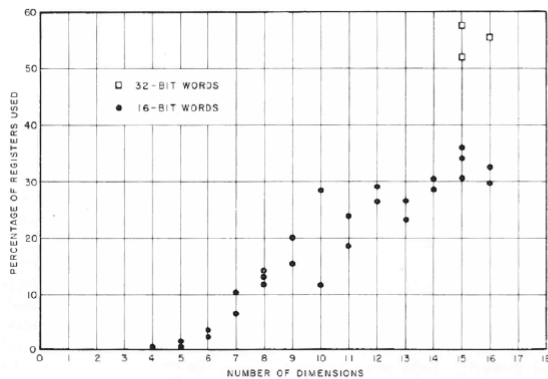


FIG. 8. Results of Monte Carlo simulation tests

the stored set. Let us examine, now, the problem of storing other arguments, focusing attention upon binary tries.

The key concept is the use of the end-of-argument marker to effect transition between argument and function. For the sake of simplicity, we may waste the right-hand cell of each end-of-argument-marking register and use only the left-hand cell. In the left-hand cell we store

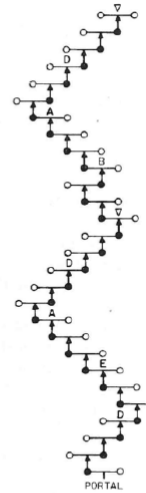


FIG. 9. Storage of a function-argument pair in a character-coded binary tree.

Capability of Trie Memory to Remove Redundancies from the Information Stored

(He explains: “The test program was written in FORTRAN for the IBM 709. The program is composed of 42 subroutines, of which 19 were coded specially for this program and 23 were taken from the library.”)

Tries didn’t make a splash when Ed first introduced them—not least because computers didn’t really have the memory then to make use of them. I think I first heard about them in the late 1970s in connection with spellchecking, and nowadays they’re widely used in lots of text search, bioinformatics and other applications.

Ed had apparently first started talking about tries when he was still in the Air Force. As he explained it to me in 2014:

The Air Force [people] had no idea [what I was talking about]. But I kept on [saying] “I need to find someone who knows something about this that can critique it for me.” And someone says to me, “There’s a guy at MIT who deals in something similar, he calls it lists”. And that was John McCarthy. So, I call up, I get a secretary and, you know, I make a date, and I go to MIT and in building 56 with the computation center, I go to his office and the secretary says he’s somewhere out in the hall. I see some guy wandering back and forth. I go up and say, “You John McCarthy?” He says, “Yes.” So, I say, “I’ve had this idea —” I can’t remember if I was in uniform or not; I might’ve been. I said, “I had this idea, and I’ve written a program and tested it. And might you take a look?” Then he takes this thing, and he starts to read it.

Then he did something that struck me as very weird. He turned around slowly and started walking away, he’s reading and walk, walk, walk, walk, stop. Turns around, walk, walk, walk, walk, back slowly, you know. Finally, he comes back and he stops and he reads and reads. And he’s obviously angry. And I thought, “This is weird.” I said “Does it make sense or anything?” He says, “Yes, it makes sense.” And I said, “Well, what’s up?” He says, “Well,

I’ve had the same idea.” And I said, “Oh.” He says, “But I’ve never written it down.” And I said, “Oh, okay. So, do you think I ought to work on it or do something?” He says, “Yeah”. So, that’s how I met John McCarthy.

Ed remained friends with McCarthy for the rest of McCarthy’s life, and involved him in many of his endeavors. In 1956 McCarthy had been one of the organizers of the conference that coined the term “artificial intelligence”, and in 1958 McCarthy began the development of [LISP](#) (which was based on [linked lists](#)). I have to say I wish I’d known Ed’s story with McCarthy much earlier; I would have handled my own interactions with him differently—because, as it was, over the course of various encounters from 1981 to 2003 I never persisted very far beyond the curmudgeon stage.

Back around 1958, the circle of “serious computer people” in the Boston area wasn’t very large—and another was [Marvin Minsky](#) (who I knew for many years). Between Ed and Licklider, both McCarthy and Minsky became consultants at BBN, and all of them would have many interactions in the years to come.

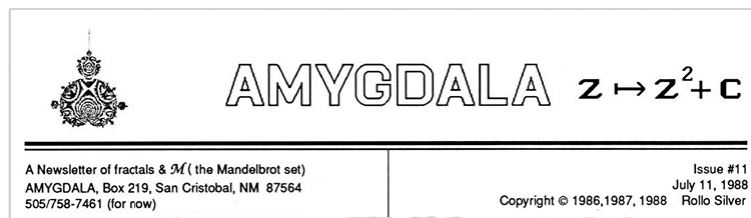
But in late 1959 there was another entrant in the Boston computer scene: the [PDP-1 computer](#), designed by a certain [Ben Gurley](#) for a new company named [Digital Equipment Corporation \(DEC\)](#) that had essentially spun off from Lincoln Lab and MIT. BBN was the first customer for the PDP-1, and Ed was its anchor user:



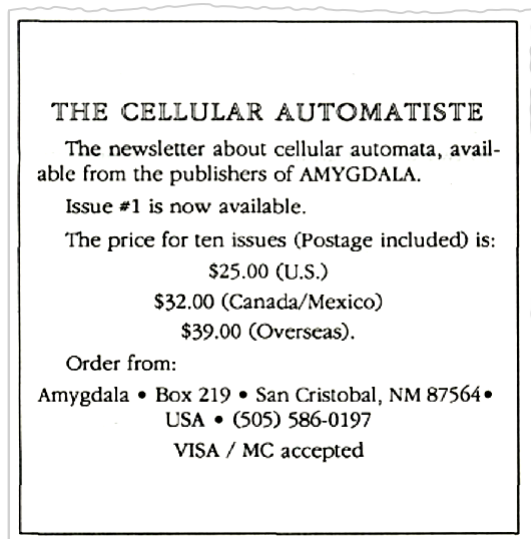
John McCarthy had had the “theoretical” [idea of timesharing](#), whereby multiple users could work on a single computer. Ed figured out how to make it practical on the PDP-1, in the process inventing what would now be called asynchronous interrupts (then the “sequence break system”). And so began a process which led BBN to become a significant force in computing, the creation of the internet, etc.

But in 1961, Ed and a certain [Roland Silver](#), who also worked at BBN, decided to quit BBN—and, strangely enough, to move to Brazil, where they were enamored of the [recently elected new president](#). But when that new president unexpectedly resigned, they abandoned their plan. And when BBN didn’t want them back, Ed decided to start a company, initially doing consulting for DEC. As Ed tells it, he and Roland Silver were such good friends and had so much they talked about that together they couldn’t get anything done, so they decided they’d better split up.

As I was writing this piece, I decided to look up more about Roland Silver—who I found out had been a college roommate of Marvin Minsky’s at Harvard, and had had a [long career in math, etc. at MITRE](#) (the holding company for Lincoln Lab). But I also remembered that many years ago I’d received letters and a [rather new-age newsletter](#) from a certain “Rollo Silver”:



Could it be the same person? Yes! And in my archives I also found an ad:



Some time after my work on cellular automata in the 1980s, Roland Silver—together with my longtime friend [Rudy Rucker](#)—started a newsletter about cellular automata, notably not mentioning Ed, but including a [colorful bio for Silver](#):

Rollo Silver, the Editor and Publisher of *The Cellular Automatiste*, is an ontological engineer who lives with his wife and works in the mountains of northern New Mexico in a house that looks like a flying saucer. Deprived of the company of his peers and half-crazed by cognitive deprivation, he started the fractal newsletter *Amygdala* in 1987.

“Triple-I” (III)

But back to Ed and his story. It was 1961, and Ed had quit his job at BBN. In 1957, he’d met on a Cape Cod beach a [woman from Western Massachusetts named Dorothy Abair](#) (who was at the time working at a beauty salon)—and six weeks later they’d married, and now had a 3-year-old daughter. Ed had already lined up some consulting with DEC, and as Ed tells it, with a little “hacking” of bank loans, etc. he was able to officially start [Information International Incorporated \(III\)](#)—with a tiny office in Maynard, MA (home of DEC). But then, one day he gets a call from the [Woods Hole Oceanographic Institute](#). He drives down to Woods Hole with a certain [Henry Stommel](#)—an oceanography professor at Harvard—who tells him about a “vortex ocean model”, and asks Ed if he can [program it on a PDP-1](#) so that it displays ocean currents on a screen. And the result is that III soon has a contract for \$10k (about \$100k today) to do this.

I might add a small footnote here. Years later I was talking to Ed about the origins of cellular automata, and he tells me that a certain Henry Stommel had told him that there were cellular automaton models of sand dunes from the 1930s. At the time—before the web—I couldn't easily track down who Henry Stommel was (and I had no idea how Ed knew him), and to this day I don't know what those sand dune models might have been.

But in any case, Ed's interaction with Woods Hole led to what became III's first major business: digital reading of film. As [Ed tells it](#):

At Woods Hole ... they had these meters which would measure how fast the ocean current was going and which way—and recorded it on 16 mm film with little tiny lights and a little fiber optic thing. And they had built a machine to read that film. I looked at the machine and said “That'll never work”. And they said “Who are you? Of course it'll work”, and so on, so forth. OK, so some months later they call me up and say it didn't work.

I have to tell you this but this is insanely funny. So I decide I'm going to make a film reader and here's how I'm going to do it. I knew there was a 16 mm projector you could rent from a company and you could stop it and then say “Advance one frame” by clicking and it would just advance one frame at a time. So I thought: say I take the lightbulb out and put a photomultiplier in and point it at the screen of the computer. Then light will come from the screen, go through the lens and be focused on the film, and some would go through the film to the photomultiplier and I would be able to tell how much light got through. And we could write a program to do the rest.

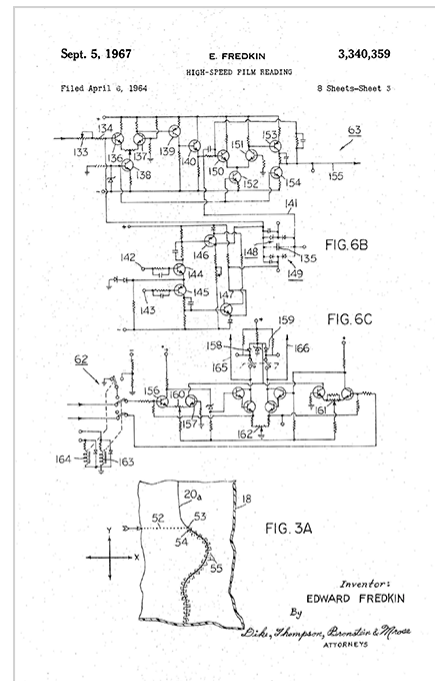
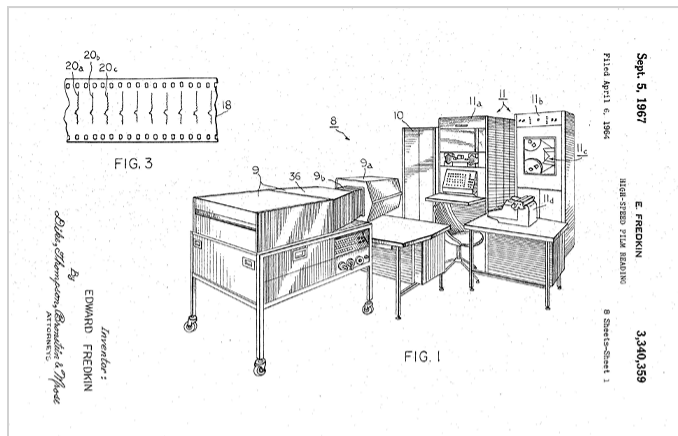
That was my idea, OK.

So not having any money, we rented that projector and I got Digital (DEC) to let me use their milling machine and I bought the photomultiplier tube, and I got Ben Gurley to design the circuitry and connect it to the computer. But there was one more thing. The photomultiplier tube was like a vacuum tube but it had like 16 pins and a very odd connector that no one had. But I thought “Lincoln Labs has parts for everything in their electronics warehouse”. So I called someone I used to work with there, and said “Look, do me a favor and sneak into the parts area, take that part and just give it to me. I've ordered one but I'm not going to get it for a while and when I get it I'll give it to you and you can put it back so it's not actually a theft.” And he said “OK, I'll do it” but he asked me why I wanted it and I told him “Well, I'm doing this stuff for Woods Hole to read some film with a computer”.

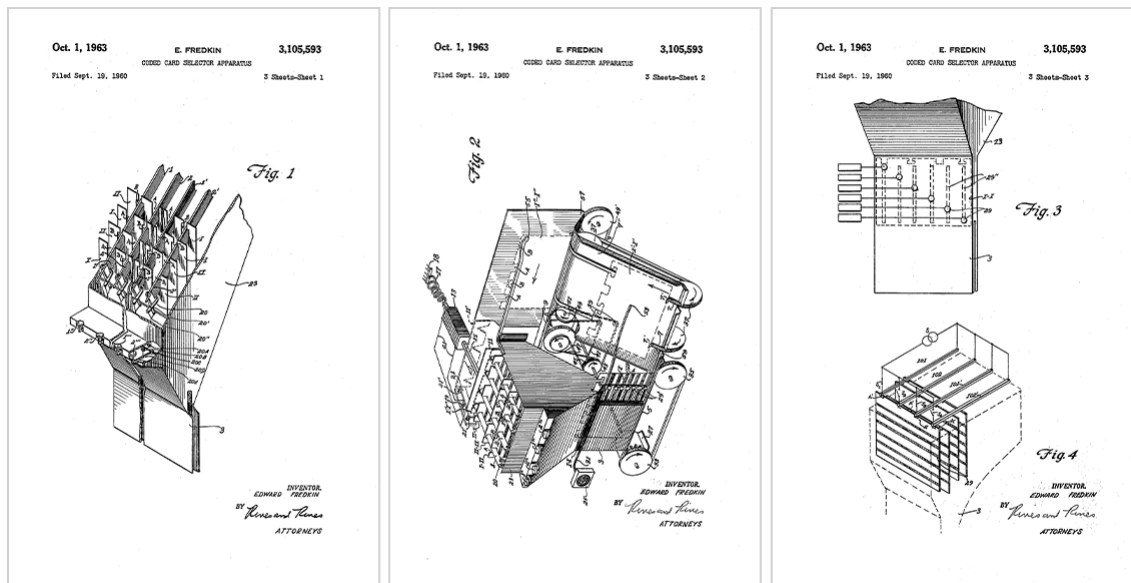
OK, so he gave me the part and we get it going right away and we're reading the film, and that solved the problem. But meanwhile this very funny thing happened. Someone from Lincoln Labs found out about all this and said “Hey, you're reading some kind of film. Is that what you used that thing for?” And I said “Yeah”. And they said “Well, we tried to read some films so we built a gadget and did the same thing you did: we pointed it at the screen of the computer, but we can't make the software work”. And I said “OK, well, come down and tell me about it”. So they come down and what happens is this. There's some army people and they have a radar that's looking at a missile coming in and records on film from

an oscilloscope. And they asked could we read this. And to make a long story short they signed another contract....

The whole setup was eventually captured in a patent entitled simply “High-Speed Film Reading”:



And actually this wasn't Ed's first patent. That had been filed in 1960, while Ed was at BBN—and it was for a [mechanical punched card sorter](#), with arrays of metal pins and the like, and no computer in evidence:



III ended up discovering that there were many applications—military and otherwise—for film readers. But their Woods Hole relationship led in another direction as well: computer graphics and data visualization. By 1963 there were perhaps 300,000 oceanographic stations recording their data on punched cards, and the idea was to take this data and produce from it a “[computer-compiled oceanographic atlas](#)”. The result was a paper:

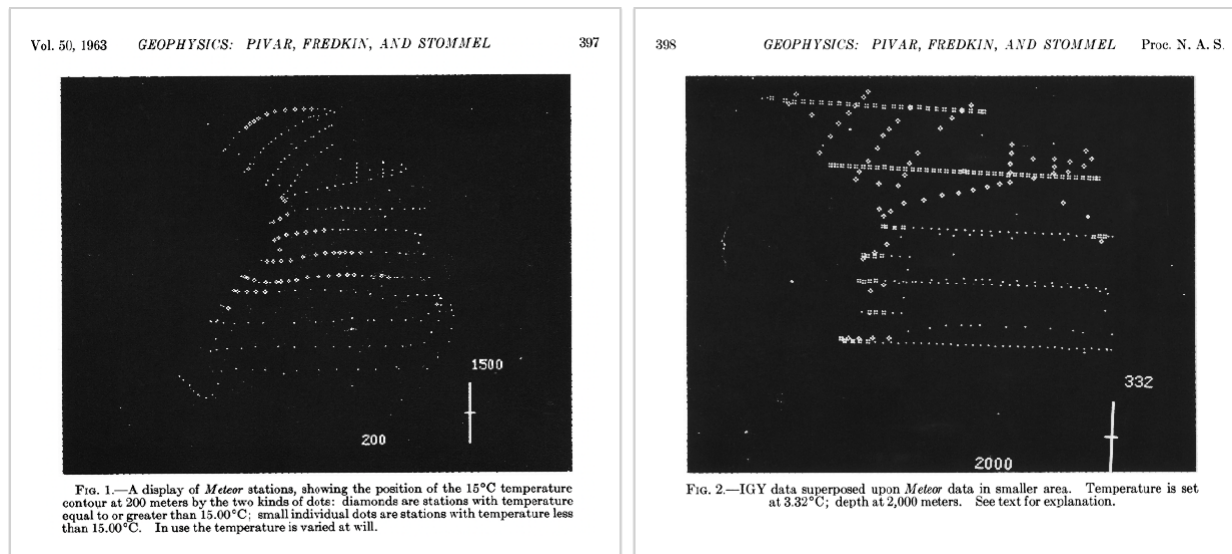
**COMPUTER-COMPILED OCEANOGRAPHIC ATLAS: AN EXPERIMENT
IN MAN-MACHINE INTERACTION***

BY MALCOLM PIVAR, ED FREDKIN, AND HENRY STOMMEL

INFORMATION INTERNATIONAL, MAYNARD, MASSACHUSETTS, AND
MASSACHUSETTS INSTITUTE OF TECHNOLOGY

Communicated June 24, 1963

And with statements like “Only a high-speed computer has the capacity and speed to follow the quickly shifting demands and questions of a human mind exploring a large field of numbers” the paper presented visualizations like:



These various developments put III in the center of the emerging field of film-meets-computers systems. The company grew, moving its center of operations to Los Angeles, not least to be near the [Systems Development Corporation \(SDC\)](#) which RAND had spun off as its software arm in response to the SAGE project.

But Ed was always having new ideas for III, and defining new directions. Ed had brought Minsky and McCarthy into III as board members and consultants, and for example in 1964 III was [proposing to SDC](#) a project to make a new version of LISP (and, yes, with no obvious film-meets-computers applications). The proposal gives some insight into the state of III at the time. It says that “From a one-man operation [in 1962], I.I.I. has grown to the point where our gross volume of business for 1964 is in the neighborhood of \$1 million [about \$10 million today]”. It explains that III has four divisions: Mathematical and Programming Services, Behavioral Science, Operations, and “New York”. It goes on to list various things III is doing: (1) LISP; (2) Inductive Inference on Sequences; (3) Computer Time-Sharing; (4) Programmable Film Readers; (5) The World Oceanographic Data Display System; and (6) Computer Display Systems.

It’s certainly an eclectic collection, reflecting, as such things often do, the character of the company’s founder. From a modern perspective, one item that catches one’s attention is:

(2) Inductive Inference on Sequences

I.I.I. has performed research on the automation of the use of inductive inference as a means of predicting the behavior of sequences of symbols or numbers. Our work in this area has ranged from theoretical examinations of methods of inference to the implementation of specific techniques on a computer, using LISP. The inductive inference programs are still under development; however, they are already sufficiently sophisticated to predict the next member of a sequence such as

0 0 0 4 0 6 0 8 9 10 0 12 0 14 15 16 0 18 0 20 21 22 ...

in which each prime number has been replaced by zero. They can also predict the next member of more straightforward series such as

Z Y X U V W T S R ...

(The next letter should be O.)

The programs can handle sequences of symbols that can be integers, real numbers, letters, or nonsense syllables. At present, there are two possible output forms: a LISP function that fits the sequence and can be used to predict an arbitrary future member, or a prediction of the next member. The programs are capable of either using or ignoring the conventional meanings of the symbols that they are presented with; those programs that ignore conventional meanings of symbols are first "primed" with collections of sequences that can be thought of as having come from previous experience. The programs will make whatever use they can of these "previous experiences" in their predictions.

One can think of it as an early attempt at AI/machine learning—which 60 years later still hasn't been solved. (GPT-4 says the next letter should be Q, not O.)

But distractions or not, it was a talented team that assembled at III—with lots of cross-fertilization with MIT. III's business progressively grew, and perhaps it outgrew Ed—and in 1965 Ed stepped down as CEO. In 1968 he left entirely and (as we'll discuss below) went to MIT, leaving III in the hands of [Al Fenaughty](#), who, years later (and after nearly 30 years at III), would become the chairman of [Yandex](#).

As someone who's curious about the ways of company founders, I asked Ed many times about his departure from III. He usually just said: "I had a partner who died". But it's only now that I've pieced together, partly from my 2014 oral history with Ed, what happened. Ed described it to me as the greatest tragedy of his life.

Shortly after he set up III, Ed persuaded Ben Gurley (designer of the PDP-1) to leave DEC and join him at III. I think Ed had hoped to build computers at III, with Gurley as their designer. But on November 7, 1963, in Concord, MA, just a few miles from where I am as I write this, Ben Gurley was murdered—by a single revolver shot through his dining room window as he was about to sit down for dinner with his wife and 7 children. An engineer from DEC (and Lincoln Labs)—about whom Gurley had recently complained to the police—was arrested, and eventually convicted of the crime (after Ed hired a private detective to help). It later turned out that a few years earlier the same engineer was likely also responsible for shooting (though not killing) another engineer from DEC.

I had always assumed that Ed's decision to leave III happened just after his “partner had died”. But I now realize that Gurley's death early in the history of III caused III to go on its path of making things like film readers, rather than the DEC- or IBM-challenging computers I think Ed had hoped for.

Even after Ed left active management of III, he was still its chairman. And in late 1968 something would happen that would change his life forever. Taking tech companies public on the “over-the-counter” market had become a thing, and a broker offered to take III public. And on December 2, 1968, III [filed its SEC paperwork](#):

INFORMATION INTERNATIONAL FILES FOR OFFERING AND SECONDARY. Information International, Inc., 545 Technology Square, Cambridge, Mass. 02139, filed a registration statement (File 2-30853) with the SEC on November 26 seeking registration of 250,000 shares of common stock with warrants to purchase 125,000 common shares, to be offered for public sale in units, each consisting of two common shares with a warrant to purchase one additional share. Of the 250,000 shares included in the units, 240,000 are being offered by the company and 10,000 (being outstanding shares) by the present holder thereof. The offering is to be made through underwriters headed by Putnam, Coffin & Burr-Doolittle, Inc., Six Central Row, Hartford, Conn. 06103, which will receive \$20,000 for expenses; the offering price (\$61 per unit maximum*) and underwriting terms are to be supplied by amendment.

The company's principal activity is to develop, assemble and market a variety of computer-based electronic-optical systems capable of dealing directly with information presented in visual form. Its principal product to date is a programmable film reader, which is a high-speed system for measurement and analysis of photographic images. At October 31, the company had no film readers on order, and a new product--the FR-80 microfilm recorder--makes up a substantial part of the company's backlog. Net proceeds of its sale of additional stock will be allocated to the costs of the company's proposed optical character recognition project; the balance will be maintained as working capital for general corporate purposes. The company has outstanding 2,306,748 common shares, of which Edward Fredkin, board chairman, owns 57.7%, Alfred L. Fenaughty, president, 12.7% and Robert W. Waller, vice president, 12.1%. The Edward Fredkin Charitable Foundation proposes to sell all of 10,000 shares held.

III's “principal product to date” is described as a “programmable film reader”, but the paperwork notes that as of October 31, 1968, the company has no film readers on order—though there are orders for its new microfilm reader, which it hasn't delivered yet. It also says that proceeds from the offering will be used to fund its “proposed optical character recognition project”. But for our purposes what's perhaps more significant is that the

paperwork records that Ed owns 57.7% of the company, with the Edward Fredkin Charitable Foundation owning 0.4%.

On January 8, 1969, III went public, and Ed was suddenly, at least on paper, worth more than \$10M (or more than \$80M today). Two years later (perhaps as soon as a lockup period expired), Ed cashed out, with the SEC notice indicating that Ed would be “repaying personal indebtedness to a bank incurred by him for reasons unrelated to the company or its business” (presumably a loan he’d taken out before he could achieve liquidity):

***EXCHANGE OFFER OF INFORMATION INTL. PROPOSED.** Information International, Inc., 12435 W. Olympic Blvd., Los Angeles, Calif. 90064, filed a registration statement with the SEC on March 29 seeking registration of 625,000 outstanding shares of common stock. Edward Fredkin, board chairman and holder of 48.1% of the company's outstanding common stock, proposes to offer to exchange these shares owned by him for cash and presently outstanding warrants (expiring January 1972, and to purchase stock at \$33 per share) in ratios to be specified later. Putnam, Coffin, Doolittle, Newburger Division of Advest Co., has agreed to head a group of dealers who will solicit acceptance of the exchange offer.

The company is engaged in developing, assembling and marketing a variety of computer-based electronic-optical systems. Net proceeds to Fredkin will be applied directly to repay his personal indebtedness to a bank incurred by him for reasons unrelated to the company or its business. (File 2-39804)

So now Ed—at age 37—was wealthy. And in fact the money he made from III would basically last the rest of his life, even through a long sequence of subsequent business failures.

III’s OCR project was never a great success, but III became a key company in digital-to-film systems (relevant to both movies and printing), and in the early 1970s created some of the very first computer-generated special effects, that eventually made it into movies like *Star Wars*. III’s stock price hovered around \$10 per share for years, and in 1996—after PostScript had pretty much taken the market for prepress printing systems—III was [sold to Autologic](#) for \$35M in stock, then in 2001 Autologic was sold to [Agfa](#) for \$42M.

The Island

When III went public in 1969 it was the height of the Cold War (which probably didn’t hurt III’s military sales). And many people—including Ed—thought World War III might be imminent. And so it was that in 1970 Ed decided to buy an island in the Caribbean, close enough to the tropics, he told me subsequently, that, he assumed (incorrectly according to current models), radioactive fallout from a nuclear war wouldn’t reach it.

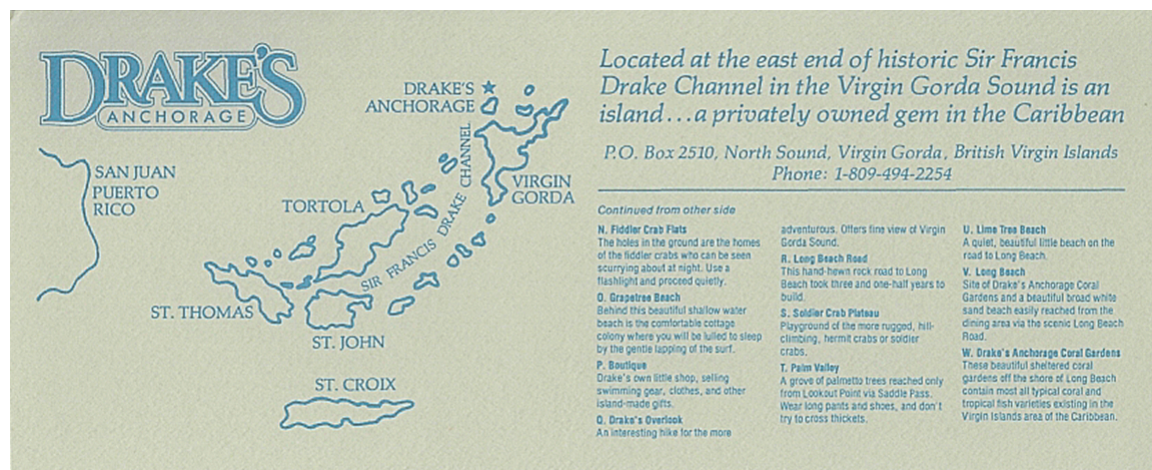
Apparently Ed was sitting in a dentist’s office when he saw an “Island for Sale” ad in a newspaper. The seller was a shipwreck-scavenging treasure hunter named [Bert Kilbride](#)—sometimes called “the last pirate of the Caribbean”—who had started to develop the island

(and for several years would manage it for Ed). It's a fairly small island (about 125 acres, or 0.2 square miles)—in the British Virgin Islands. And its name is **Mosquito Island** (or sometimes, with some historical justification, Moskito Island). And when Ed bought it, it probably cost something under \$1M. (Richard Branson bought the nearby but smaller **Necker Island** in 1978.)

I visited Ed's island in January 1982—the first time I met Ed. And, yes, there was a certain “lair of a Bond villain” (think: *Dr. No*) vibe to the whole thing. Here are pictures I took from a boat leaving the island (notice the just-visible seaplane parked at the island):



There was a small resort (and restaurant) on the island, named Drake's Anchorage (built by the previous owner):



And, yes, there were beaches on the island (though I myself have never been much of a beach-goer):



And, in keeping with the Bond vibe, there was a seaplane too:



There was one house on the island, here pictured from the plane (it so happened that when I visited the island, I was learning to fly small planes myself—so I was interested in the plane):



Visiting a [nearby island](#)—with its very rundown airport sign—gives some sense of the overall area:



Ed claimed it was difficult to run the resort on his island, not least because, he said, “the British Virgin Islands have the lowest average worker productivity in the world”. But he nevertheless, for example, had a functioning restaurant, and here I am there in 1982, along with Charles Bennett, about whom we’ll hear more later:



When people talked about Ed, his island was often mentioned, and it projected a general image of overall mystique and extreme wealth. In 1983 a movie called *WarGames* came out, featuring a reclusive military-oriented computer expert named “Professor Falken”—who had an island. Many people assumed Falken was based on Fredkin (and it now says so all over the internet). However, in writing this piece, I decided to find out what was actually true—so I asked one of the writers of the movie, [Walter Parkes](#). He responded, and, yes, fact is often even stranger than fiction:

Unfortunately I can confirm that Ed was not the inspiration for Stephen Falken. The character was inspired by Steven [sic] Hawking. (Falken = Falcon = Hawking) The movie was first conceived to be about two characters, a young super-genius born into a family incapable of acknowledging his gifts, and a dying scientist in need of a protégé. In the first several drafts Falken was confined to a wheel-chair and was working on understanding the big bang, for which he had created a computer simulation. Little known fact—while writing the character, we had one person in mind to play the role: John Lennon, who was murdered shortly before we finished the script.

(By the way, in a moment of “fact follows fiction”, *WarGames* featured a computer with lots of flashing lights. I happened to see the movie with [Danny Hillis](#), and as we were walking out of the movie, I said to Danny “Perhaps your computer should have flashing lights too”. And indeed flashing lights became a signature feature of Danny’s [Connection Machine](#) computer, as later seen in movies like *Jurassic Park*.)

Project MAC

After he left III in 1968, Ed's next stop would be MIT, and specifically [Project MAC](#) (the "Multiple Access Computer" Project). But actually Ed had already been involved much earlier with Project MAC. In many ways the project was a follow-on to what Ed had been doing at BBN on timesharing.

In 1963 Ed wrote a long survey article on timesharing:

THE TIME SHARING OF COMPUTERS

*Edward Fredkin
Information International, Inc.
Maynard, Mass.*

Time sharing may be described as the simultaneous use or sharing of a computer by a number of users.

In a typical time-sharing system, the users may communicate with a central computer by means of a number of individual consoles (usually consisting of typewriters or teletypewriters). They may prepare and put in programs just as on an ordinary computer. For example, in one type of time-sharing system, the central computer will run each user's program, in turn, for a specified "quantum" of time (e.g., 20 milliseconds). At the end of that time the computer stores the user's program in a readily accessible pe-

ripheral memory device, such as a disc file. It reads in the next user's program from peripheral storage, processes it for 20 milliseconds, re-stores it, proceeds to the next user, and so on until, in round-robin fashion, it has returned to the first user. The first user's program will then receive a second "quantum" of processing, and the cycle will continue as before. Many other time-sharing methods besides this one are, however, now in use.

To understand time sharing and its promises for the future, let us first pay some attention to its history.

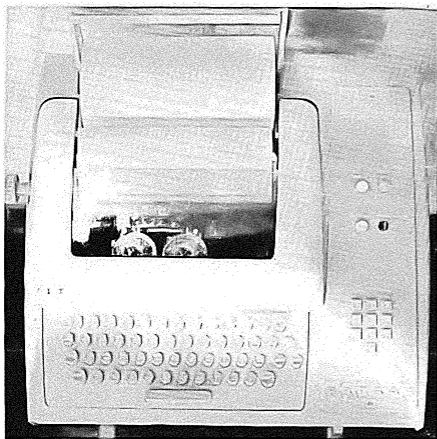
A Short History of Time Sharing

In the Fifties it was clear to many computer people that

- (1) computers would eventually think,
- (2) the main problems would lie in knowing what to do and how to do it,
- (3) computers would help us to do it,
- (4) that doing what we decided to do would be a programming task,
- (5) that computers would help us do the programming tasks, and
- (6) that we needed systems and languages that allowed us to use the capabilities of computers to the extent of our needs.

Our first main problem was an economic one. We could see that researchers would benefit from continuous access to the most capable (and most expensive) machines. Yet no one was about to give a new IBM 704 to each person interested in contributing to science, although there were impressive arguments to the effect that it would be worth doing this for at least some people!

Strangely enough, however, all the needs of one individual could be met by only a small part of a big machine's capabilities provided these capabilities were distributed (or shared) properly in time. This problem had been attacked and solved in the area of allowing input-output devices to share the time of the central processor—Wesley Clark of Lincoln Laboratory of Mass. Inst. of Technology not only designed a most impressive input-output interrupt system into the TX-2 computer, but he clearly specified in 1954, in a memo entitled "The Multi-Sequence Program Concept," exactly how to write a subroutine capable of simultaneous execution by many different callers. This



Typical input/output station for an individual user under a time sharing system.

The introduction contains a rather charming window onto the view of computers at the time:

A Short History of Time Sharing

In the Fifties it was clear to many computer people that

- (1) computers would eventually think,
- (2) the main problems would lie in knowing what to do and how to do it,
- (3) computers would help us to do it,
- (4) that doing what we decided to do would be a programming task,
- (5) that computers would help us do the programming tasks, and
- (6) that we needed systems and languages that allowed us to use the capabilities of computers to the extent of our needs.

And the ads interspersed through the article give a further sense of the time:

CONTROL DATA CORPORATION TO SCIENTIFIC & INDUSTRIAL MANAGEMENT

THIS IS TO INFORM YOU THAT CONTROL DATA CORPORATION ANNOUNCES THE CONTROL DATA® 3200 COMPUTER SYSTEM, NOW AVAILABLE AS THE INDUSTRY'S FASTEST, MOST VERSATILE, LEAST EXPENSIVE MEDIUM-SCALE COMPUTER SYSTEM ANYWHERE TODAY.

3200 DESIGN IS SPECIFIC TO YOUR ORGANIZATION'S REQUIREMENTS WITH DECIMAL ARITHMETIC TO PROCESS MANAGEMENT DATA, FLOATING POINT FOR HIGH PRECISION SCIENTIFIC COMPUTING, MODULAR INPUT-OUTPUT FOR COMPREHENSIVE REAL-TIME PROBLEMS OR ANY OPTION COMBINATION FOR MOST COMPLETE SYSTEM MANAGEMENT.

FIRST DELIVERIES SCHEDULED TO AUSTRALIA COMMONWEALTH BUREAU OF CENSUS AND STATISTICS TO PROCESS DATA ON POPULATION, DEMOGRAPHIC TRENDS, SOCIO-ECONOMIC STATISTICS, GOVERNMENT FINANCIAL ACCOUNTING. ALSO TO COMMONWEALTH SCIENTIFIC AND INDUSTRIAL RESEARCH ORGANIZATION (CSIRO) FOR RESEARCH IN ASTROPHYSICS, CHEMICAL ENGINEERING, AGRICULTURAL SCIENCES, RESEARCH SUPPORT TO AUSTRALIAN UNIVERSITIES AND GOVERNMENT SCIENTIFIC GROUPS.

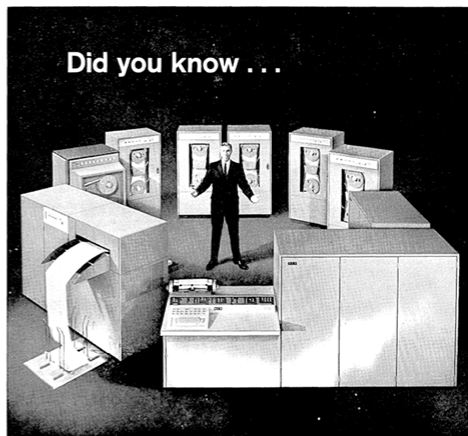
COMPLETE 3200 INFORMATION AVAILABLE NOW, NEAREST CONTROL DATA REPRESENTATIVE.

Sales Offices: Albuquerque • Birmingham • Boston • Chicago • Cincinnati • Dallas • Denver • Detroit • Houston • Los Angeles • London • Madrid • Mexico City • New York • Philadelphia • Pittsburgh • Portland • St. Louis • St. Paul • Toronto • Washington, D.C. • Zurich

CONTROL DATA CORPORATION



Did you know . . .



you can get an NCR 315 Computer System for only \$3800 per month?

YES . . . five magnetic tape handlers, a punched paper tape reader, a high-speed printer, a console, and a processor containing a 10-thousand character memory for only \$3800 per month.

With this NCR 315 computer system more businesses than ever before can enjoy the benefits of electronic data processing. It is a system that is expandable—up to 16 NCR CRAM (Card Random

Access Memory) files, up to four magnetic character sorters, up to three more magnetic tape files, and a punched card input/output unit can be added as future requirements dictate.

There's more we could say about this system . . . space does not permit. Give your local NCR Representative a call . . . he has all the facts.

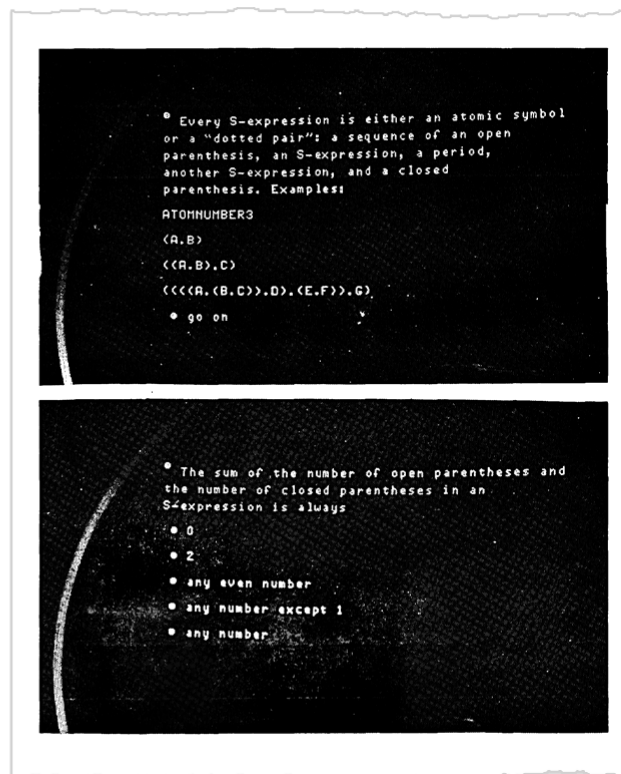
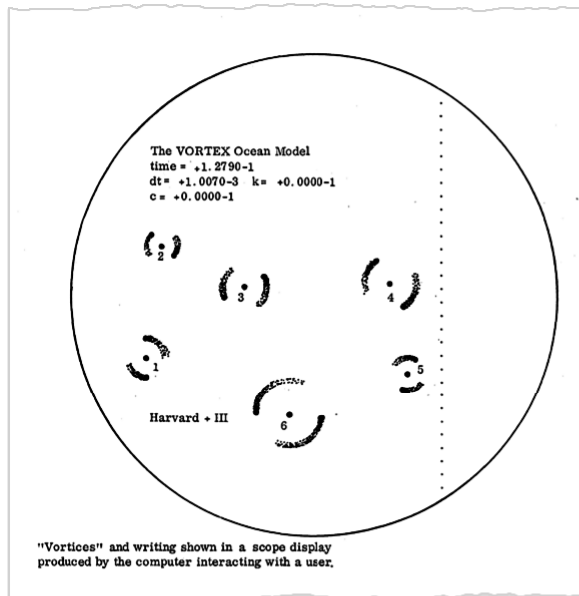
NCR PROVIDES TOTAL SYSTEMS—FROM ORIGINAL ENTRY TO FINAL REPORT—THROUGH ACCOUNTING, REPORTING, DATA REGISTERS ON ACTIONS, MAINTENANCE, AND DATA PROCESSING. The National Cash Register Co.-1233 offices in 120 countries—70 years of helping business save money.

NCR

COMPUTERS and AUTOMATION for November, 1965

21

As illustrations of what can be done with an interactive timeshared computer, there's a picture from Ed's vortex ocean simulation—as well as an example of an online “book” about LISP:



And, yes, already a kind of “cloud computing” story:

Time sharing should make feasible the use of a computer as a programmed teaching machine. Above are two sample frames as appearing on the scope of a PDP-1. If 2000 users shared a computer renting for \$1 million a year and each station was used for 2000 hours a year, the hourly cost for each user would be 25 cents.

There's also a description of Project MAC—that had just been funded by the [Advanced Research Projects Agency \(now DARPA\)](#). The article said that the “MAC” stood either for “Multiple Access Computer” or “Machine-Aided Cognition”. It included various sections on what might be possible with timesharing:

Intercommunication

One of the benefits of a time-sharing system may be to allow a highly useful intercommunication among the users of the system. For example, it could provide general message routing and distribution. Messages would be distributed from one user to another by name, distribution list, specific user category, or other more complicated criteria. An additional advantage of a time-sharing system may lie in its ability to provide person-to-person communication on an “information retrieval” basis. For example, the system could have available information that answers questions such as “Who should I contact to find out this information?” It could accept a question and transmit it to the person mentioned. It could delay transmission until the person is present. It could accept a reply which is delayed until the receiver is present. It could, in other words, serve as a programmable delayed intercommunication net.

The main text of the article ends with a rousing (?) vision of AI taking over from humans (and, yes, even though this is from 60 years ago it’s not so different from what at least some people might say about the “AI future” today):

The Big Picture of the Far Future

If we assume that the development of artificial intelligence will take place, how can we form a coherent picture of our place and its place in the great scheme of things?

It seems reasonable to consider that artificial intelligence is a next step in the evolutionary process. Once there was a prototype land creature that crawled out of some primeval ocean; most primitive in comparison with a more advanced creature such as a sabre-tooth tiger—and once there was a prototype intellectual creature (that’s us), and we will seem most primitive in comparison with what is to come! However, instead of participating in evolution by merely serving as a runner in a genetic message relay race, we can choose the great distinction of directly employing our primitive yet unique intellectual talent to the task of evolution.

We have conceivably arrived at the notion of the “next step in evolution” by trying to fit the future development of Artificial Intelligence into the over-all scheme of things. Joshua Lederberg, a Nobel prize-winning geneticist at Stanford University, gave some attention to the problem of guessing the characteristics of that species that will evolve from mankind. He arrived at the conclusion that it would be Artificial Intelligence.

How do we proceed with this great task? J. C. R. Licklider’s position in 1959 was that we should concentrate on the development of Man-Computer Symbiosis,⁶ a system in which men and machines would form a close-knit intellectual team, each complementing the capabilities of the other. Then, when so equipped, we could much more effectively tackle Artificial Intelligence.

After the ascendancy of A.I., it seems that mankind’s intellectual skills will no longer be very useful in the solution of problems. Today, man is hardly using the intellectual capability he has. How ironic it would be to pass through our span as the dominant species in Earth’s evolutionary tree, without ever realizing our true potential for intellectual creativity!

By means of a symbiotic relation with the capabilities that we now know how to implement in computers, we can not only proceed with the task of creating A.I., but we can reach for and achieve human creativity at a pace and level never before attained. We could advance all areas of science and art as partners in intellectual systems that complement and reinforce our correctable weaknesses and that make use of all our inherent creativity.

With computable wings we could soar to meet the Phoenix.

But there’s a curious piece of backstory to Project MAC—from 1961—that appears as a footnote to Ed’s article:

⁹ Marvin Minsky and the author, one evening in 1961 in Pasadena, California, called on Richard Feynman, physicist at California Institute of Technology. We had a far-ranging discussion that evening—settling on how computers could be better than pencil and paper in aiding mathematicians in symbolic manipulation. We all felt that this was a good idea, and in the course of the evening (and early morning) developed in some detail the concept of the Mathematical Laboratory. Since that time, the three of us have actively pursued this goal. Feynman has worked with Glenn Culler, of the University of California at Santa Barbara, in areas associated with numerical and graphical techniques. And our organization, Information International, has developed a wide range of techniques in the field of number theory. See “Computer Aids to Number Theory,” by Malcolm Pivar (to be published in the Fall 1963 Proceedings of the DECUS users group).

Ed told me versions of this story many times. McCarthy had failed to get tenure at MIT, and was looking for another job. (Yes, in retrospect this seems remarkable given all the things he’d already done by then. But those things were computer science—and MIT didn’t yet have a CS department; McCarthy was in the EE department.) Ed, Minsky and McCarthy were going to an SDC meeting in Los Angeles, and while he was out there McCarthy was going to interview at Caltech (his undergraduate alma mater). They had a free evening, and Ed suggested they meet “someone interesting”. Ed remembered Linus Pauling from his time at Caltech. But Pauling wasn’t in. So Minsky suggested they call [Richard Feynman](#). And he was in, and invited them over to his house.

Feynman apparently showed them things like his nanotech-inspiring tiny motor, etc., but somehow the discussion shifted to AI. And Minsky mentioned work a [student of his](#) was doing on the “AI problem” of symbolic integration. Then McCarthy started to explain ways a computer could do algebra. Then, as Ed told it to me in 2014:

Feynman produces this sheaf of papers to show us. It was all algebra. And he says “There’s a problem. I’ve done this calculation, and it’s close to 50 pages. A graduate student has done it too, and Murray Gell-Mann has done it. And the only thing we know for sure is that our three results are mutually inconsistent. And the only conclusion we can arrive at is that a person can’t do this much algebra with the hope of getting it right.” And so the question was could there be some system that could help do a problem like that? So what happened is Marvin [Minsky] and I basically fleshed out the idea of a mathematical thing. And it was agreed that we would do it. Marvin and I decided to divide this task up, that I would do one part, and he would do another. Now, we had one bad idea in there, OK. It’s partly Feynman’s fault, but it’s also Marvin and my fault. He was convinced you could not do [math] by typing it. It had to have some kind of handwriting recognition. So, it was decided I would do the handwriting recognition...

And although I didn't know this until I was writing this piece, it turns out the original proposal for Project MAC was actually based on the idea of building a system for mathematics, and "Project MAC" was originally the "Project on Mathematics and Computation". Pretty soon, though, the emphasis of Project MAC would shift to the "infrastructure" of timeshared computing. But there was still a math effort, which in time became the [MACSYMA](#) system for computer algebra (written in LISP by students and grandstudents of Minsky).

And here this intersects with my personal story. Because many years later (starting in 1976) [I would use that system](#)—along with other early computer algebra systems—to do all sorts of physics calculations. My archives still contain an example of what it was like in 1980 to log in to "Project MAC" over the ARPANET (my username was "swolf" in those days; note the system message, the presence of 15 MITishly-named "lusers" altogether, and yes, mail):

```
RAND TIP 424 #: 3
@o 236
Trying...
Open T
Open R
Macsyma Consortium KL-10

MC ITS.1192. FWORD.1733.
TTY 51
15. Lusers, Fair Share = 49%
```

```
As we all know, MC is very loaded during the afternoon hours. It is
becoming necessary for us to restrict usage during those hours to only
Consortium users (users with group U) and LCS members, and to exclude
all other users (users with group N) from noon to 6:00 PM Eastern
time. This will shortly be implemented in software, but in the
meantime we are asking those people with flexible schedules to spread
themselves around the clock a bit more. Morning hours, e.g. from 5:00
am to about 10:00 am, are still quite unloaded. In addition, we would
like all LCS/AI members to avoid the noon to 6:00 PM time period if
they possibly can do so.
```

```
*swolf$u
Password:
[OK]
```

```
THURSDAY, JULY 24, 1980 00:47:46 AM
Welcome to ITS
```

```
--Mail--
Date: 21 JUL 1980 1934-EDT
From: KED at MIT-MC (Keith Ellery Dow)
To: HEP at MIT-MC
```

But, actually, in late 1979 I had already decided to "do my own thing" and build my own system for doing mathematical computation, and eventually much more. And indeed when I first met Ed in 1982 I had recently finished the [first version of SMP](#), and to commercialize it I had started my first company. In 1986 I started to build [Mathematica](#) (and what's now

[Wolfram Language](#))—which was released in 1988. Ed started using Mathematica very soon after it was released, and basically continued to do so for the rest of his life.

But picking up the original Project MAC narrative from 1963: the old group from BBN had dispersed but were still [writing together about timesharing](#) (and when they said a “debugging system” they meant essentially what we would now call an operating system):

**A TIME-SHARING DEBUGGING SYSTEM
FOR A SMALL COMPUTER**

J. McCarthy
Computation Center, Stanford University, Stanford, California
S. Boilen
Bolt Beranek and Newman Inc., Cambridge, Mass.
E. Fredkin
Information International Inc., Maynard, Mass.
J. C. R. Licklider
Advanced Research Projects Agency, Department of Defense

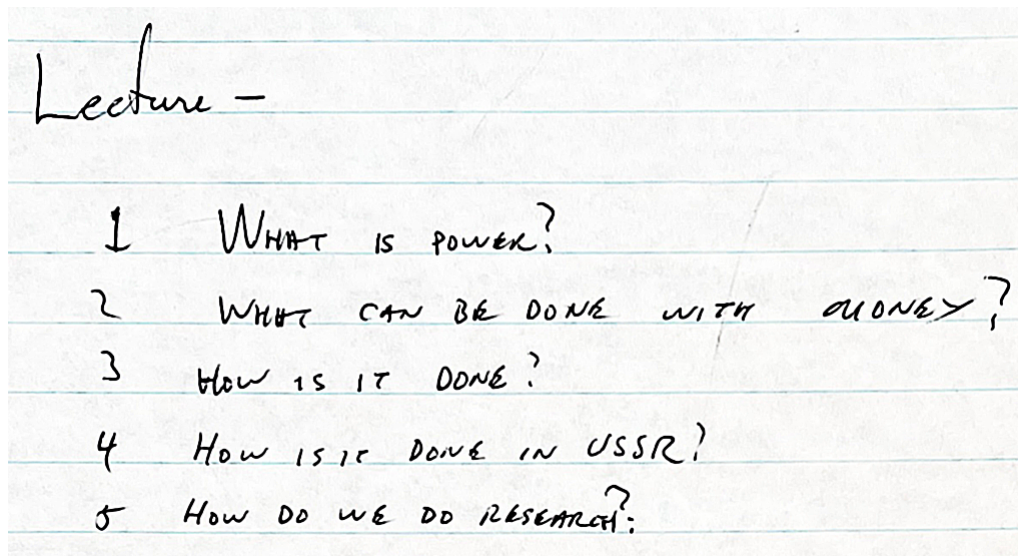
And when Project MAC launched in 1963, its “steering committee” included Minsky, Gurley—and Ed. (John McCarthy had landed at Stanford, where he would remain for the rest of his life. I first met him in 1981, at a time when Stanford was trying to recruit me. There was a lunch with the CS department; people went around the room and introduced themselves. McCarthy unhelpfully—and confusingly—said he was “John Smith”.)

Ed at MIT

In 1968, Ed left III—and Minsky, together with Licklider (who had by then become director of Project MAC), persuaded the MIT EE department to hire Ed as a visiting professor for the year. Ed had been spending most of his time at III in Los Angeles, but III also had a *pied-à-terre* in the Boston area, and indeed its IPO documents listed its address as 545 Technology Square, Cambridge—the very building in which Project MAC was located.

At MIT, Ed invented and taught a freshman course on “Problem Solving”. He told me many times one of his favorite “problem exercises”. Imagine there’s a person who can cure anyone who’s sick just by touching them. How could one set things up to make the best use of this? I must say I never find such implausible hypotheticals terribly interesting. But Ed was proud of a solution that he’d come up with (I think in discussion with Minsky and McCarthy) that involved systematically shuttling millions of people past the healer.

This probably didn't come from that particular course, but here are some notes I found in an archive of Ed's papers at MIT that perhaps suggest some of the flavor of the course (we'll talk about Ed's interest in the Soviet Union later):



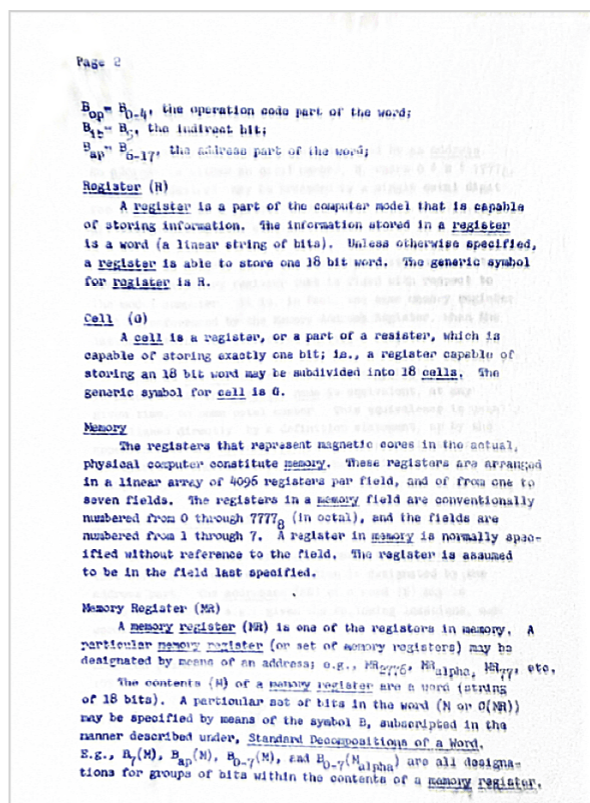
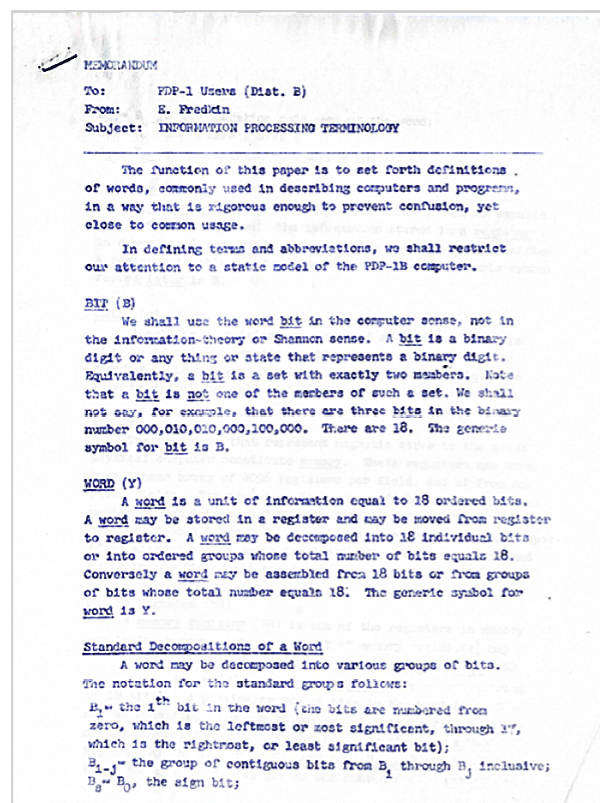
In 1968 MIT—and Project MAC in particular—was at the very center of emerging ideas about computer science and AI. A picture from that time captures Ed (third from left) with a few of the people involved: [Claude Shannon](#), [John McCarthy](#) and [Joe Weizenbaum](#) (creator of [ELIZA](#), the original chatbot):



At the end of the 1968 academic year student reviews from Ed's course were unexpectedly good, and MIT needed faculty members who could be principal investigators on the government grants that were becoming plentiful for computing—and one of those typical-for-Ed “surprising things” happened: MIT agreed to hire him as a full professor with tenure, despite his lack of academic qualifications. It was a watershed moment for Ed, and I think a piece of validation that he carried with pride for the rest of his life. (For what it's worth, while Ed was an extreme case, MIT was at that time also hiring at least some other people without the usual PhD qualifications into CS professor positions.)

In 1971 Licklider stepped down from his position as director of Project MAC—and Ed assumed the position. His [archives from the time](#) contain lots of administrative material—studies, reports, proposals, budgets, etc.—including many pieces reflecting things like the birth of the ARPANET, the maturing of operating systems and the general enthusiasm about the promise of AI.

One item (conceivably from an earlier time) is Ed's summary of “Information Processing Terminology” for PDP-1 users, complete with definitions like: “A bit is a binary digit or any thing or state that represents a binary digit. Equivalently, a bit is a set with exactly two members. Note that a bit is not one of the members of such a set”:



Ed does not seem to have been very central to the intellectual activities around Project MAC, and the emerging Lab for Computer Science and AI Lab. But his name shows up from time to time. And, for example, in the classic “[HAKMEM](#)” collection of 191 math and CS “hacks” from the AI Lab, there are two—both very number oriented—attributed to Ed:

ITEM 58 (Fredkin):
 $3^3 + 4^3 + 5^3 = 6^3$.

ITEM 38 (Fredkin):
 By the binomial theorem, the b th forward difference at 0 of
 the 0, 1, 2, ... powers of n is $(n-1)^b$. E.g., for $n = 4$:

1	4	16	64	256
3	12	48	192	
9	36	144		
27	108			
81				

In fact, any straight line with rational slope through such an array will always go through a geometric sequence with common ratio of the form $n^a (n-1)^b$. In the above, east by southeast knight's moves give the powers of 12: 1, 12, 144, ...

Rollo Silver gets mentioned too—notably in connection with “random number generators” involving XORs (and, yes, the code is assembly code—for a PDP-10):

ITEM 25 (Schroeppel):
 Random number generators, such as Rollo Silver's favorite, which use SHIFTS and XORs, and give as values only some part of their internal state, can be inverted. Also, the outputs may often be used to obtain their total internal state. For example, 2 consecutive values from Rollo's suffice to allow prediction of its entire future. Rollo's is:

```

RANDOM: MOVE A,HI      ;register A gets loaded with "high" word
        MOVE B,LO      ;register B gets loaded with "low" word
        MOVEM A,LO      ;register A gets stored in "low" word
        LSHC A,35.      ;shift the 72-bit register AB left 35
        XORB A,HI      ;bitwise exclusive-or of A and HI
                        replaces both
  
```

This suggests a susceptibility to analysis of mechanical code machines.

Also in HAKMEM is the “munching squares” algorithm—that I was later shown by [Bill Gosper](#):

```

ITEM 146:
Another simple display program: ("munching squares")
It is thought that this was discovered by
Jackson Wright on the RLE PDP-1 circa 1962.
    DATAI 2
    ADDB 1,2
    ROTC 2,-22
    XOR 1,2
    JRST .-4
2=X, 3=Y. Try things like 1001002 in data switches. This also
does interesting things with operations other than XOR,
and rotations other than -22. (Try IOR; AND; TSC; FADR; FDV(!);
ROT -14, -9, -20, ...)

ITEM 147 (Schroeppel):
Munching squares is just views of the graph  $Y = X \text{ XOR } T$ 
for consecutive values of  $T = \text{time}$ .

```

And talking of Gosper (whom I've known since 1979, and who almost every week seems to send me mail with a surprising new piece of math he's found with Mathematica): in 1970 the [Game of Life cellular automaton](#) had come on the scene, and Gosper and others at MIT were intensely studying it, with Gosper triumphantly discovering the glider gun in November 1970. Curiously—in view of all his emphasis on cellular automata—Ed doesn't seem to have been involved.

But he did do other things. In 1972, for example, as a kind of spinoff from his Problem Solving course, he [formed a group called "The Army to End the War"](#) (i.e. the Vietnam War), whose idea was that it was time to stop the government fighting an unwinnable war, and this could be achieved by having an organization that would coordinate citizens to threaten a run on banks unless the war was ended. Needless to say, though, this didn't really fit well with the project Ed ran being funded by the Department of Defense.

Between MIT being what it is, and Ed being who he was, there were often strange things that happened. As Ed tells it, one day he was in Marvin Minsky's office talking about unrecognized geniuses, and a certain [Patrick Gunkel](#) walks in, and identifies himself as such. Ed ended up having a long association with Gunkel, who produced such documents as:

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

CAMBRIDGE, MASSACHUSETTS 02139

"DEUS EX MACHINA?"

_____, _____, in room _____,
Project MAC sponsors a panel discussion by
Professor Edward Fredkin, Professor Terry
Winograd, & Dr Pat Gunkel on the outre theme
of the conception of a mechanical civiliza-
tion in the next century and beyond.

The dense and colorful conversation will
twistingly navigate

- 1) Human potentiality, finiteness, lia-
bility, and aspirations,
- 2) Technical limits, bases, qualities,
& problems of new beings,
- 3) Reasons for and against such creatures,
- 4) Transitional reactions and interactions
of machines and men,
- 5) Opera and existence of automorphic
supermachines,
- 6) A summary vista.

The panel is lambent if latrant & inturnd
if interpellable. She shall attempt profund-
ity, truth, beauty, and invention. The am-
bience is both canny and translunary. The
discussion is for _____ minutes and an overrun.
Everybody is enthusiastically invited. Informa-
tion, Professor Fredkin.

(Gunkel's major goal was to create what he called "ideonomy", or the "science of ideas", with divisions like isology, chorology, morology and crinology. I met Gunkel once, in Woods Hole, where he had become something of a local fixture, riding around town with his cat in his bicycle basket.)

But after a few years as director of Project MAC, in 1974 Ed was onto something new: being a visiting scholar at Caltech. After his 1961 encounter, he had gotten to know Richard Feynman—who always enjoyed spending time with "out of the box" people like Ed. And so in 1974 Ed went for a year to Caltech, to be with Feynman.

The Universe as a Cellular Automaton

My own efforts (and successes) with cellular automata may perhaps have had something to do with it. But I think at least in the later part of his life, Ed felt his greatest achievements related to cellular automata and in particular his idea that the universe is a giant cellular automaton. I'm not sure when Ed really first hatched this idea, or indeed started to think about cellular automata. Ed had told me many times that when he'd told John McCarthy "the idea", McCarthy suggested testing it by looking for "roundoff error" in physics,

analogous to roundoff error from finite precision in computers. Ed scoffed at this, accusing McCarthy of imagining that there was literally “an IBM 709 computer in the sky”. And Ed’s implication was that he had gotten further than that, imagining the universe to be made more abstractly from a cellular automaton.

I didn’t know quite when this exchange with McCarthy was supposed to have taken place (and, by the way, some of the emerging experimental implications of our [Physics Project](#) are precisely about finding evidence of discrete space through something quite analogous to “roundoff errors” in the equations for spacetime). But Ed’s implication to me was always that he’d started exploring cellular automata sometime before 1960.

In the mid-1990s, researching history for my book *A New Kind of Science*, (as I’ll discuss below) I had a detailed email exchange and long phone conversation with Ed about this. The result was a statement in my [notes about the history of cellular automata](#):

page 908.) Around 1961 Edward Fredkin simulated the 2D analog of rule 90 on a PDP-1 computer, and noted its self-reproduction properties (see page 1179), but was generally more interested in finding simple physics-like features.

At the time, Ed made it sound very convincing. But in writing this piece, I’ve come to the conclusion it’s almost certainly not correct. And of course that’s disappointing given all the effort I put into the history notes in my book, and the almost complete lack of other errors that have surfaced even after two decades of scrutiny. But in any case, it’s interesting to trace the actual development of Ed’s ideas.

One useful piece of evidence is a 25-page document from 1969 [in his archives](#), entitled “Thinking about New Things”—that seems to outline Ed’s thinking at the time. Ed explains “I am not a Physicist, in fact I know very little about modern physics”—but says he wants to suggest a new way of thinking about physics:

P.R. Klein
Recd. Sep 3, 1969

E. Fredkin - 1969
"Thinking About New Things"

notion *minds*
 The purpose of this paper is to plant a ~~bug~~ in the ~~end~~ of today's
 physicists and philosophers about new ways of thinking about things.
 The ideas that I shall present are in many ways too premature for
 publication, however they bear on so much and in such a major way that
 I feel that it is imperative for the best of scientists to start thinking
 about the ~~possibilities~~ possibilities. It has occurred to me to work on
 this theory until it was solved and proven, and to publish an
 important paper, but, for various reasons I am not working on this theory
 with anything like the energy that it deserves. In addition I am not a
 Physicist, in fact I know very little about modern physics (XX I am sure
 that any competent physicist will be able to poke various holes in my
 exposition, however I hope that they will not long focus on the errors,
 but try and understand the nature of the general scheme and what it might
 mean)

Soon he starts talking about the possibility that the universe is “merely a simulation on a giant computer”, and relates a version of what he told me about his interaction with John McCarthy:

After I became a competent computer programmer, I often speculated on the philosophical implications of computers. One day in 1959 I was talking to ^{a friend} ~~John McCarthy~~ about things, and ~~xxxx~~ since he was much more knowledgeable than I about things, it occurred to me to ask him questions about simulation. "If this universe were merely a simulation on a giant computer in some other universe, would there be any way for us to detect

that fact?" John replied that the thought had already occurred to him, and he saw no reason why it would be difficult to prove that our universe was a simulation.

"We could look for round-off or ~~truncation~~ truncation error," ^{said, John.} ~~he replied,~~ I knew that I was on to something good, since John had thought of it first.

While the thought of a giant IBM in the sky seemed somewhat ludicrous, there were certain attractive aspects. What such a giant computer would be doing depended on your school of philosophy. For example, perhaps I am the only reality, and the computer is simply pretending to be me. Whenever I look in a particular direction the computer quickly begins to simulate just that part of the universe that I am cognizant of. The computer does not have to bother itself with anything other than creating a plausible state for anything that I sense. ^{follows} If we ~~ascribe to the~~ more current vogue in philosophy, then the computer would be very busy computing the interactions amongst all of the fundamental particles. Even this computation can be done in many ways, for example, ~~as~~ the computer ~~could~~ set aside a few registers for each particle, to represent ^{this} its momentum and position, and it ~~could~~ compute for each quanta of time, the interactions ^{among} of the particles. On the other hand the computer could solve the universe as a giant mesh, where each point ^{a lattice-like} in space was represented by a few ~~but~~ words of memory (in terms of field and perhaps particles) and for each quanta of time the machine would have to compute the interaction of each point with its immediate neighborhood.

He talks (in a rather programmer kind of way) about the beginning of the universe:

The attractive aspects that I mentioned previously are that a few puzzling questions ^{suddenly} become easy to answer. First, how did the universe begin? That's easy; the computer was loaded with the initial state of the universe and then the start button was pushed. This does not beg the question, since there would be no need to assume that the universe in which the computer existed ~~has~~ had problems about beginnings and ends, had conservation laws, etc. In fact it seems reasonable to assume that it is different than this one, for why else would they go to the trouble to simulate us? (Actually one can think of reasons why)

He goes on—again in a charmingly “programmer” way:

So far, we see ^{over} that the model of the universe that compares it to a computer simulation ^{has} some attractive features. However it really is sort of absurd to think of the giant IBM in the sky. Needless to say there are many unattractive aspects to such a model. The major flaw in the model is ~~xxx~~ that it is hopelessly contrived. What I mean is that the idea of trying to determine if the great computer is more like the IBM 360, or like the Russian BESM-6 is sillier than most of the similar kinds of questions that puzzled early religious philosophers. The goal of a new physics should be to introduce clarity and simplicity, the giant IBM serves to add questions of round-off and truncation error to all of the existing problems of modern physics.

For a while I speculated about the possibility of a cute resolution of the computation error dilemma, ~~H~~ GRAVITY might be the error in the computation of basic electromagnetic interaction. ~~H~~ This was appealing since gravity was a very weak force (only about 10^{-40} as strong as the electrostatic forces). It was easy to see that one could write a program to compute electromagnetic interaction, and to contrive to have the resulting errors show up as a consistent central attractive force,

by juggling the precision of the computation steps, and by juggling the way in which the arithmetic was performed, it would be easy to produce ~~XXXXXXXXXX~~ gravity that was arbitrarily close in magnitude to the observed real gravity. Of course, the goal would be to find a simple and untampered with algorithm that land exactly on the right value, rather than to slowly creep towards the correct value by having more and more complex rules for the nature of the computation.

Even so, there was not much benefit to be gained from finding such a model, since it would not have the order and simplicity that are demanded of revolutionary ideas.

A bit later, Ed is beginning to get to the concept of cellular automata:

Once the advantages of a ~~phy~~ digital or discrete model for physics became more app^earant to me I began to ~~try~~ try to think of simpler ideas and mechanisms. If such a model could be developed, then algorithms involving bits of information w~~u~~xx would become ~~h~~ the basic ingredi^ents of all processes that we observe, and we would be able to achieve a total and complete understanding of the nature of the most basic aspects of physics. The reason for this is that the idea of a finite state automata is exact and easy to understand. It is a process that involves information in a simple yet precise way.

Let us consider a simple automata which I will refer to as a cell. The cell has exactly two internal states. Unlike a complex thing such as an atom, which can be in one of many states, our simple cell will have just two. In that sense it is much like an electron that can have one of two spins. We could name these two states anything that we want, such as "on, off" or "up, down" or "true, false" but I will call them one and zero (1, 0). The interesting part is the rule that allows or causes a cell to change from one state to another.

The rule might be as follows "at each instant of time (each tick of the clock) the cell looks at its inputs and becomes some simple function of its inputs, and its present state." Let us also assume that the inputs are also just ones or zeros. One specific rule might be "The new state is that state held by a majority of the inputs. Another rule might be "The new state is to be "one" if an odd number of the inputs are "one", otherwise it is to be zero." Another rule might be to ignore the inputs and to set the state to zero.

The important nature of such a device is that the description is complete and total, nothing will ever affect the outcome of the operation of such a cell other than the inputs and the particular rule in effect.

And there we have it: Ed gets to (3D) cellular automata, though he calls them “spatial automata”:

Such a cell is not very interesting by itself, however if we arrange a large number of them in some orderly fashion, then interesting things begin to happen.

Let us imagine that we have a space made up of a cartesian lattice of points. Each point would have exactly six neighbors (unless it was on an edge or in a corner). We can again think of the neighbors as *up, down, north, south, east and west*. If we imagine that each such point in a very large space of points was a cell of the type that was previously described, and that the inputs to each cell were the states of itself and its six neighbors, then we would have a spatial automata of a most interesting type. What is so interesting is the richness of behaviour that results from the simplest of rules. Such spatial automata can exist in any number of dimensions, and many of the most important observations about their behaviour are independent of the number of dimensions. In all subsequent discussion we will assume that the lattice of cells stretches infinitely in all directions, so that we will not have to worry about ~~some~~^{the} effects along the boundaries.

And now he claims that spatial automata can exhibit “very complex behavior”—although his meaning of that will turn out to be a pale shadow of what I [discovered in the early 1980s with things like rule 30](#):

A spatial automata is certainly one of the simplest structures that one can imagine that is capable (as we shall show) of exhibiting very complex behaviour.

But at this point Ed already seems to think he’s almost there—that he’s almost reproduced physics:

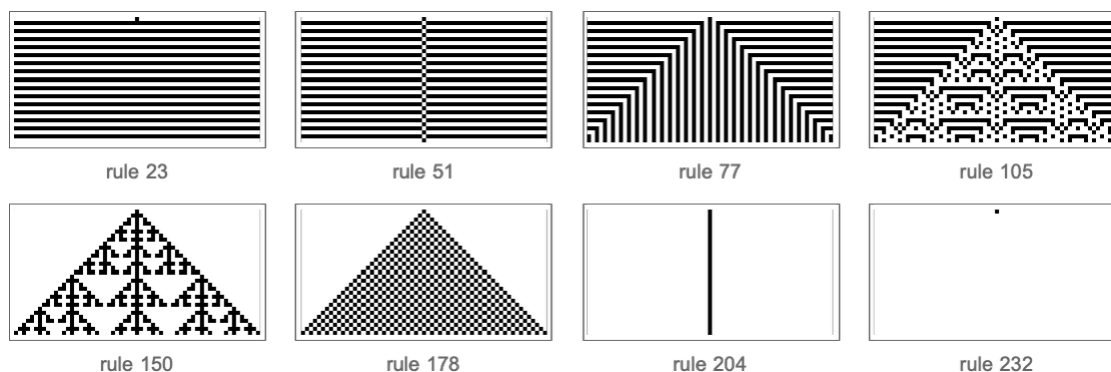
A little later he's discussing doing something very much in my style: enumerating possible rules:

And still further on he actually talks about 1D rules. And in some sense it might seem like he's getting very close to what I did in the early 1980s. But his approach is very different. He's not doing "science" and "empirically seeing what cellular automata do". Or even being very interested in cellular automata for their own sake. Instead, he's trying to engineer cellular automata that can "be the universe". And so for example he wants to consider only left-right symmetric cellular automata "because the universe is isotropic". And having also decided he wants cellular automata that are symmetric under interchange of black and

white (a property he calls “syntactic symmetry”), he ends up with just 8 rules. He could just have simulated these by running them on a computer. But instead he tries to “prove” by pure thought what the rules will do—and comes up with this table:

RULE	NAME	STABLE STATES	META STABLE STATES (TEMPORARY STABILITY)	LONGEST PERIOD	LONGEST APERIODIC SEQUENCE
000	NOR	none	none	2	1
001	LEFT SHIFT	...000000....	...100100010000100..	1	n
010	NO SHIFT	isolated <u>ones</u>	none	1	1
011	LEFT CLEAR	all <u>zeros</u>	none	1	2
100	RIGHT SHIFT	all <u>zeros</u>	same as LEFT SHIFT	1	n
101	ALTERNATE	all <u>zeros</u>	...1010101010101...	1	n/2
110	RIGHT CLEAR	all <u>zeros</u>	none	1	2
111	AND	all <u>zeros</u>	...1111111111111...	1	n/2

Had he done simulations he might have made pictures like these (labeled using my rule-numbering scheme):



But as it was he didn't really come to any particular conclusion, other than what amount to a few simple “theorems” about what “data processing” these cellular automata can do:

Some important generalizations are: For all rules other than the NOR rule, all zeros is stable. Rules 001, 010, and 100 will ~~maintain~~ maintain a constant number of ones. (After the first step, and in an infinite or ring type system) Rules 011, 101, and 110 will decrease the number of ones until there are none, in ~~one~~ two steps, while 111 does the same in at most $n/2$ steps. If a rule consists of a function that cannot increase the number of ones, such as LEFT CLEAR, then its period cannot be greater than 2, and the total length of the longest aperiodic behaviour will be n .

I must say I find it very odd that—particularly given all the stories about his activities and achievements he told me—Ed never in the four decades I knew him mentioned anything about having thought about 1D cellular automata. Perhaps he didn't remember, or perhaps—even after everything I wrote about them—he never really knew that I was studying 1D cellular automata.

But in any case, what comes next in the 1960 document is Ed getting back to “pure thought” arguments about how cellular automata might “make physics”:

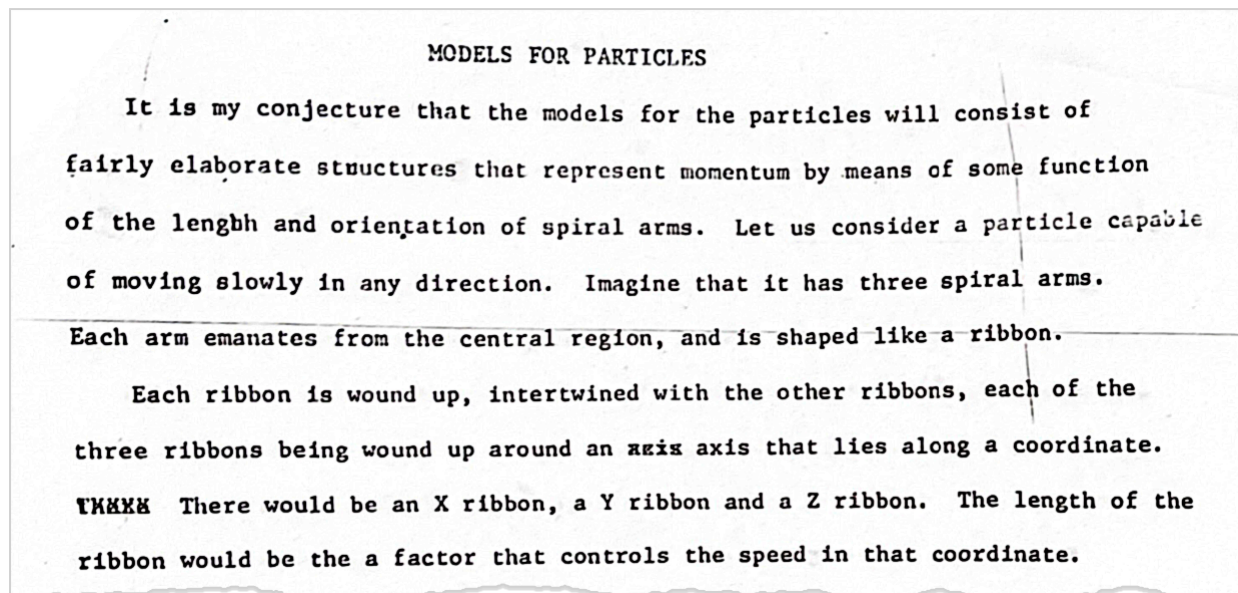
JUST HOW SPATIAL AUTOMATA MIGHT MODEL PHYSICS

Let us consider a simple particle such as an electron, and see if we can imagine a structure of ones and zeros in some spatial automata that behaves as does an electron. The ~~fixx~~ first property that we need to ~~xxxxx~~ explain is the conservation ~~law~~ of electrons law. What this means, is that aside from certain exceptions electrons tend to be conserved. Thus if we have a pattern of bits that represents an electron, it should not spontaneously dissolve into ~~xxxxxx~~ nothing, nor should such a pattern appear from nowhere. Of course, ~~ix~~ if the electron should meet up with a positron, then the two should do what they're supposed to do. We know from previous examples that in a spatial automata with a rule such as exclusive-or, that many kinds of patterns are quite stable. Some types of patterns seem to propagate like waves, and others seem to exhibit oscillatory motion. In addition many ~~kinkx~~ kinds of simple structures seem to have a fleeting existence as a cohesive entity while in the process of decay. All of ~~xxx~~ this kind of observation leads one to imagine that certain very large kinds of structures could have the necessary stability to ~~x~~ retain their identity indefinitely. Such structures could have the property of being able to move through the space relative to other such structures, and could interact with other similar and different stable structures, and in addition could interact with various kinds of fields.

If one can imagine that a very large assembly of ones and zeros, within the space of a spatial automata of the type that we have describe, can retain its identity and in addition exhibit certain properties, then it becomes possible to think of models for the fundamantal particles. Such models of particles and fields would be extremely clear and easy to understand. The interactions of such particles would be increasingly complex as more and more of the cells would be needed to contain the interacting particles. Any simple physics experiments could be conducted by means of either a computer simulation using existing computers, or by an experiment on an exact hardware ~~xxxxx~~ analogy of the spatial ~~xxxxxxx~~ automata. It is suggested that experiments conducted with digital hardware thatx simulates exactly the rules of physics could substantially replace experiments done with particle acceleraters.

It's a bit muddled (though, to be fair, this was a document Ed never published), but at the end it's basically saying that if the universe really is just a cellular automaton then one should be able to replace physical experiments (that would, for example, need particle accelerators) with “digital hardware” that just runs the cellular automaton. The next section is entitled “The Design of a Simulator”, and discusses how such hardware could be constructed, concluding that a $1000 \times 1000 \times 1000$ 3D grid of cells could be built for \$50M (or nearly half a billion dollars today).

After that, there's one final (perhaps unfinished) section that reads a bit like a caricature of “I've-got-a-theory-of-physics-too” mechanical models of physics:

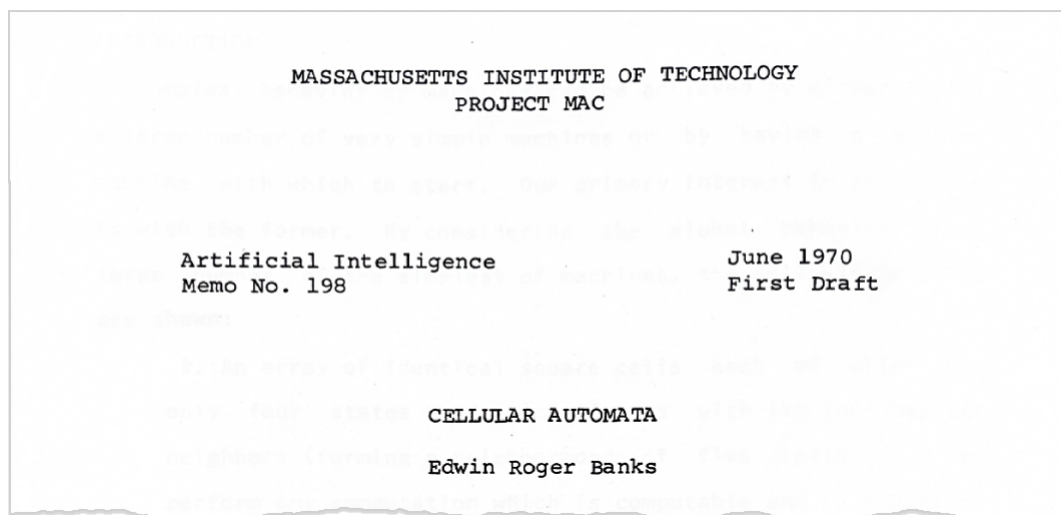


But, OK, so what does this all mean? Well, first, I think it makes it rather clear that (despite what he told me) by 1969—let alone 1961—Ed hadn't actually implemented or run cellular automata in any serious way. It's also notable that in this 1969 piece Ed isn't using the term “cellular automaton”. The [concept of cellular automata had been invented many times](#), under many different names. But by 1969 the term “cellular automaton” was pretty firmly established, and in fact 1969 might have represented the very peak so far of interest in cellular automata in the world at large. But somehow Ed didn't know about this—or at least wasn't choosing to connect with it.

Even at MIT [Frederick Hennie](#) in the EE department had actually been [studying cellular automata](#)—albeit under the name “iterative arrays”—since the very beginning of the 1960s. In 1968 [E. F. Codd](#) from IBM (who laid the foundations for SQL—and who worked with Ed's friend [John Cocke](#)) had published a book entitled *Cellular Automata*. [Alvy Ray Smith](#)

—in the same department as John McCarthy at Stanford—was writing his PhD thesis on “cellular automata”. In 1969 Marvin Minsky and Seymour Papert published their *Perceptrons* book, and were apparently talking a lot about cellular automata. And for example by the fall of 1969 Papert’s student Terry Beyer had [written a thesis](#) about the “recognition and transformation of figures by iterative arrays of finite state automata”—under the auspices of Project MAC, presumably right under Ed’s nose. (And, no, the thesis doesn’t mention Ed, though it mentions Minsky.)

Right around that time, though, something happens. Ed had been convinced—probably by Minsky and McCarthy—that any cellular automaton capable of “being the universe” better be computation universal. And now there’s a student named [Roger Banks](#) who’s working on seeing what kind of (2D) cellular automaton would be needed to get computation universality. Banks had found examples requiring much fewer than the 29 states von Neumann and Burks had used in the 1950s. But—as he related to me many times—Ed challenged Banks to find a 2-state example (“implementable purely with logic gates”), and Banks soon found it, first describing it in June 1970:



Banks had apparently been interacting with the “Life hackers” at MIT, and in November 1970 some of the thunder of his result was stolen when Bill Gosper at MIT discovered the glider gun, which suggested that even the rules of the Game of Life (albeit involving 9 rather than 5 2D neighbors) were likely to be sufficient for computation universality.

But for our efforts to trace history, Banks’s June 1970 report has a number of interesting elements. It relates the history of cellular automata, without any mention of Ed. But then—in its one mention of Ed—it says:

Considering only self-reproducing patterns, Edward Fredkin has described the following interesting cellular space. If the states are "0" and "1" and if the result state is obtained by taking the sum of the neighbors, modulo 2, then any initial configuration will reproduce copies of itself symmetrically about the initial configuration. Terry Winograd generalized this result showing that any neighborhood, not necessarily just the four nearest neighbors, and any number of dimensions still give the same results. Further, if there are p states 0, 1, ..., $p-1$ where p is a prime number, then the sum of the neighbors modulo p is a rule that will assure self-replication of the pattern in a finite number of time steps.

The “mod-2 rule” that Ed told me he’d simulated in 1961 has finally made an appearance. In an oral history years later Terry Winograd [reported that in 1970](#) he “went to a lecture of Papert’s in which he described a conjecture about cellular automata [which Winograd] came back with a proof of”.

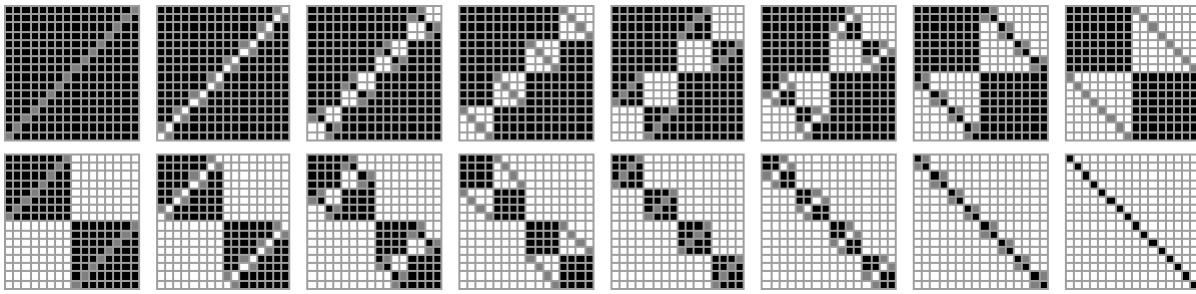
By January 1971, Banks is [finishing his thesis](#), which is now officially supervised by Ed (even though it’s nominally in the mechanical engineering department):

ACKNOWLEDGMENTS

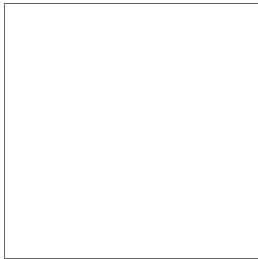
I wish to thank Professor Edward Fredkin, the supervisor of this thesis, for introducing me to the topic and for providing expert supervision especially pertaining to the direction which this investigation has followed. I wish also to thank Professor Thomas B. Sheridan, the thesis committee chairman, and Professors Marvin Minsky and Henry Paynter, the other members of my thesis

Most of Banks’s work is presented as what amount to “engineering drawings”, but he mentions that he has done some simulations. I don’t know if these included simulations of the mod-2 rule but it seems likely.

So was 1969 or 1970 the first time the mod-2 rule had been heard from? I’m not sure, but I suspect so. But to confuse things there’s a “display hack” known as “munching squares” (described in HAKMEM) that looks in some ways similar, and that was probably already seen in 1962 on the PDP-1. Here are the frames in a small example of munching squares:



Here's a video of a bigger example:



I expect Ed saw munching squares, perhaps even in 1962. But it's not the mod-2 rule—or actually a cellular automaton at all. And even though Ed certainly had the capability to simulate cellular automata back at the beginning of the 1960s (and could even have recorded videos of 2D ones with III's film technology) the evidence we have so far is that he didn't. And in fact my suspicion is that it was probably only around the time I met Ed in 1982 when it finally happened.

My First Encounter with Ed

In May 1981 there'd been a conference at MIT on the [Physics of Computation](#). I'd been invited, but in the end I couldn't go—because (in a pattern that has repeated many times in my life) it coincided with the initial release of my SMP software system. Still, in December 1981 I got the following invitation:

MASSACHUSETTS INSTITUTE OF TECHNOLOGY



Laboratory for Computer Science

M. L. Dertouzos, Director

(formerly Project MAC)
 545 Technology Square
 Cambridge, Massachusetts 02139
 (617) 253-3194
 (617) 666 8595 (home)

J. Moses, Associate Director

Tom Toffoli NE43-830

Stephen Wolfram
 Physics Dept, Caltech,
 PASADENA CA 91125

13 Dec 81

Dear Stephen,

We would like to invite you to a workshop on "Information Mechanics," to be held at the Drake's Anchorage Hotel on Mosquito Island from the 18th to the 29th of January 1982. The workshop is being organized by the MIT Information Mechanics group, and is also being sponsored by the Carnegie Mellon University.

Information Mechanics is the study of the connections between Computer Science and Theoretical Physics. Work in this area has already yielded very exciting results. Some of the people invited to this workshop have proven, to the surprise of many, that it is theoretically possible to build computers that are perfectly energy-efficient; namely they do not need to dissipate any heat or electricity! Another recent and surprising result is the modeling of general purpose computers by systems obeying the laws of Newtonian mechanics; this establishes a direct link between computation and physics where none was expected.

Information Mechanics is receiving much attention at

The workshop will involve about twelve participants and two or three students. We will have a Perq computer system available, and a special-purpose hardware simulator for cellular automata capable of supporting a wide range of rules at very high speed. The Perq will serve as an advanced office automation system, to aid in preparing position or discussion papers. In addition, we have programs on it for running the Billiard Ball model and Cellular Automata simulations. The MIT students attending the workshop will be available as programmers to quickly implement new ideas as running programs on the Perq. On most days, we plan to hold two general sessions, one in the morning and one in the late afternoon. These sessions will last about 90 minutes, and will allow for the presentation of ideas. Most discussion will take place amongst informal groups that will meet during the day. Mosquito Island is an ideal place for this workshop, completely isolated from all distractions. One of the participants will be Richard Feynman, who is recuperating from extensive surgery. Because of his condition, his ability to attend such a meeting requires the kind of setting that we have arranged.

Even so, his participation will naturally be limited to what will be feasible considering his strength and state of health.

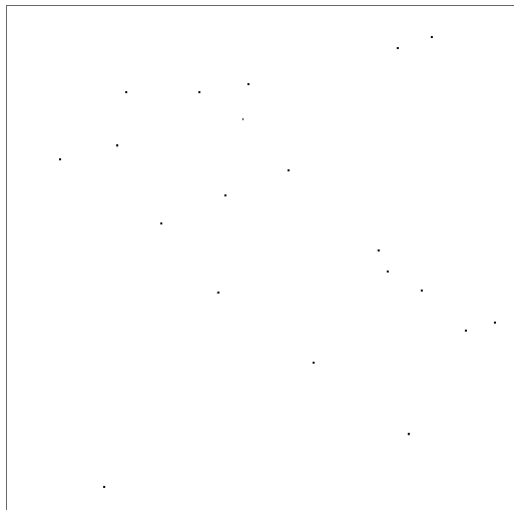
In January 1982 I was planning to go to England to do a few weeks of intensive SMP development on a computer that a friend's startup had—and I figured I would go to the Caribbean “on the way”.

It was an interesting group that assembled on January 18, 1982, on Mosquito Island. It was the first time I met my now-longtime friend [Greg Chaitin](#). There were physicists there, like [Ken Wilson](#) and [David Finkelstein](#). (Despite the promise of the invitation, Feynman's health prevented him from coming.) And then there were people who'd worked on reversible computation, like [Rolf Landauer](#) and [Charles Bennett](#). There were [Tom Toffoli](#) and [Norm Margolus](#), who had their cellular automaton machine with them. And finally there was Ed. At first he seemed a little Gatsby-like, watching and listening, but not saying much. I think it was the next morning that Ed pulled me aside rather conspiratorially and said I should come and see something.

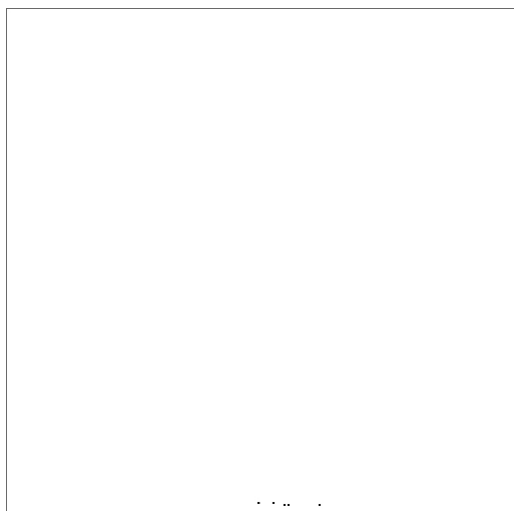
There was just one real house (as opposed to cabin) on the island (with enough marble to clinch the Bond-villain-lair vibe). Ed led me to a narrow room in the house—where there was a rather-out-of-place-for-a-tropical-island modern workstation computer. I'd seen workstation computers before; in fact, the company I'd started was at the time (foolishly) thinking of building one. But the computer Ed had was from a company he was CEOing. It was a [PERQ 1](#), made by [Three Rivers Computer Corporation](#), which had been founded by a group from CMU including McCarthy's former student [Raj Reddy](#). I learned that Three Rivers was a company in trouble, and that Ed had recently jumped in to save it. I also learned that in addition to any other challenges the engineers there might have had, he'd added the requirement that the PERQ be able to successfully operate on a tropical island with almost 100% humidity.



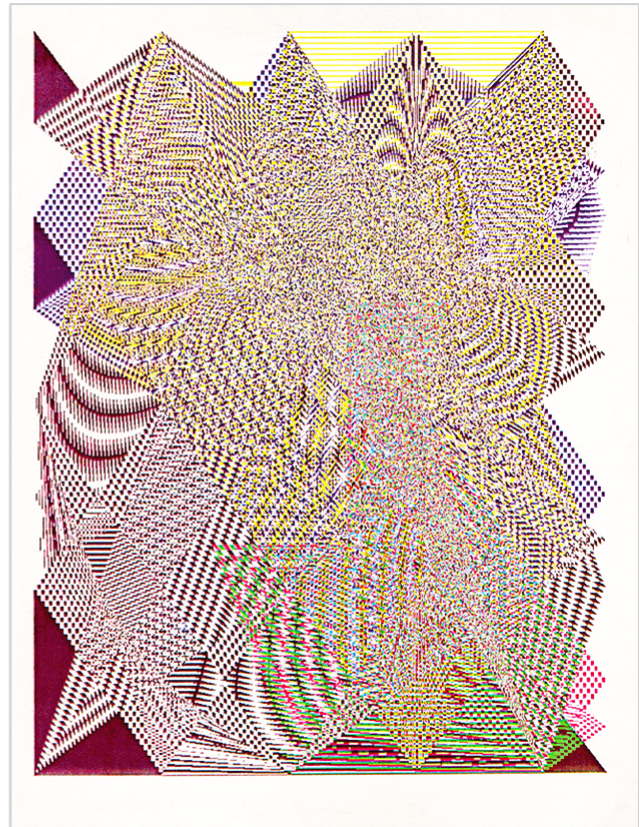
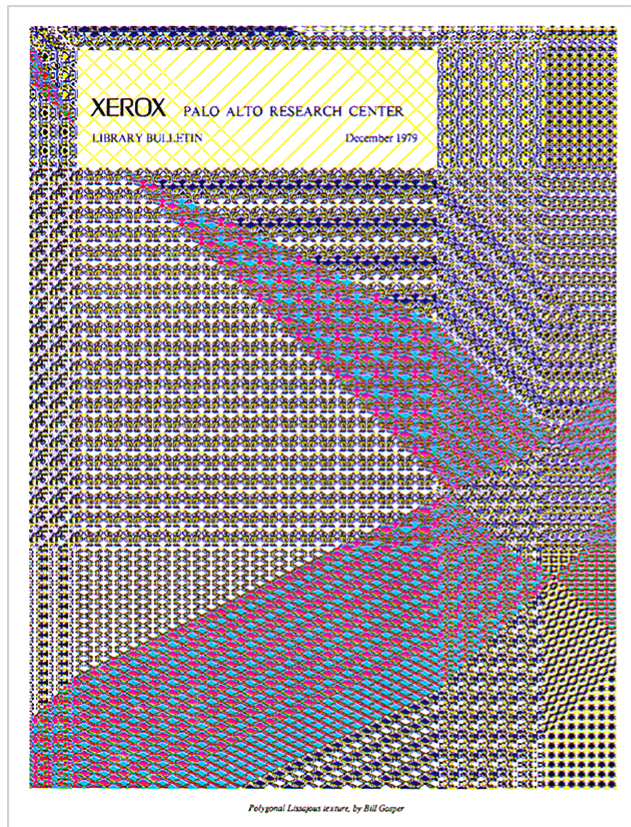
But in any case, Ed wanted to show me something on the screen. And here's basically what it was:



Ed pressed a button and now this is what happened:



I’d seen plenty of “display hacks” before. Bill Gosper had shown me ones at [Xerox PARC](#) back in 1979, and my archives even contain some of the early color laser printer outputs he gave me:



I don't remember the details of what Ed said. And what I saw looked like “display hacks flashing on the screen”. But Ed also mentioned the more science-oriented idea of reversibility. And I'm pretty sure he mentioned the term “cellular automaton”. It wasn't a long conversation. And I remember that at the end I said I'd like to understand better what he was showing me.

And so it was that Ed handed me a PERQ 8” floppy disk. And now, 41 years later, here it is, sitting— still unread—in my archives:



It's not so easy these days to read something like this—and I'm not even sure it will have “magnetically survived”. But fortunately—along with the floppy—there's something else Ed gave me that day. Two copies of a 9-page printout, presumably of what's on the floppy:


```

program CA1(input, output);
exports
  screen from screen;
const {$include TD.Dfs}
var screenp: rasterptr;
c: char;
quitting, evenstep: boolean;
param, pz, py, px, delay: integer;
function keyhit: boolean;
function r(i, j: integer): integer;
function random: integer;
function sqroot(x: integer): integer;
procedure clearscreen;

private
  imports io Unit from io Unit;
  imports ioErrors from ioErrors;
  imports ControlStore from ControlStore;
  var hitkey: boolean;
  TD, TDBin: MicroFile;

function keyhit;
begin
  if IORead(TransKey, c)=IOEIOC then begin
    if (ord('0')<ord(c)) and (ord(c)<ord('9'))
    then begin
      if hitkey then param:=0;
      param:=param*10+(ord(c)-ord('0'));
      hitkey:=false
    end else
      hitkey:=true;
    keyhit:=hitkey
  end else
    keyhit:=false
  end;

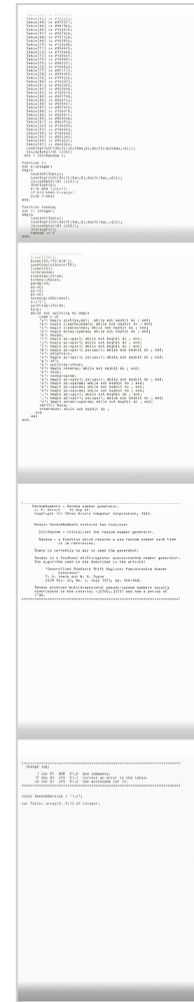
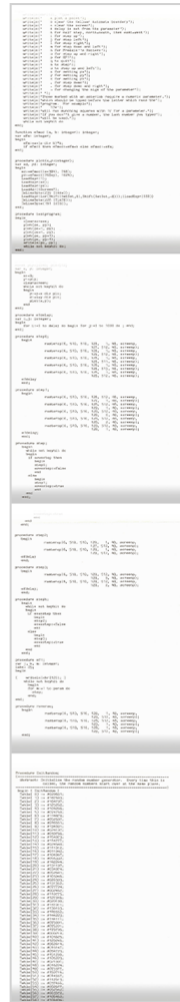
procedure clearboundary;
begin
  rasterop(6, 128, 1024, 0, 0, 48, screenp, 0, 0, 48, screenp);
  rasterop(6, 130, 1024, 638, 0, 48, screenp, 638, 0, 48, screenp);
  rasterop(6, 768, 1, 0, 0, 48, screenp, 0, 0, 48, screenp);
  rasterop(6, 768, 1, 0, 511, 48, screenp, 0, 511, 48, screenp);
  rasterop(6, 768, 2, 0, 1022, 48, screenp, 0, 1022, 48, screenp)
end;

procedure clearscreen;
begin
  rasterop(6, 768, 1024, 0, 0, 48, screenp, 0, 0, 48, screenp)
end;

function sqroot;
var v, i: integer;
begin
  LoadExpr(x);
  LoadExpr(L0r(Shift(Sqrt,8),Shift(Sqrt,-8)));
  InlineByte(191 {JCS});
  StorExpr(v);
  Sqrt:=v
end;

procedure help;
begin
  writeln(chr(12), 'Type:');
  writeln('    a plot a point');

```



And what's there is basically a **Pascal** program (and the PERQ was a very Pascal-oriented machine; “PERQ” is said to have stood for “Pascal Engine that Runs Quicker”). But what does the program do? The main program is called “CA1”, suggesting that, yes, it was supposed to do something with cellular automata.

There are a few comments:

```

{-----
RandomNumbers - Random number generator.
J. P. Strait 15 Sep 80.
Copyright (C) Three Rivers Computer Corporation, 1980.

Module RandomNumbers contains two routines:

InitRandom - initializes the random number generator.

Random - a function which returns a new random number each time
it is referenced.

There is currently no way to seed the generator.

Random is a feedback shift-register pseudo-random number generator.
The algorithm used is one described in the article:

'Generalized Feedback Shift Register Pseudorandom Number
Generator'
T. G. Lewis and W. H. Payne
JACM Vol. 20, No. 3, July 1973, pp. 456-468.

Random produces multidimensional pseudo-random numbers equally
distributed in the interval -32768..32767 and has a period of
2^98.
-----}

```

```

{-----
Change log:

1 Jun 81 BAM V1.2 Add comments.
17 Sep 80 JPS V1.1 Correct an error in the table.
20 Jun 81 JPS V1.2 Use microcode for TD.
-----}

```

And there's code for making help text:

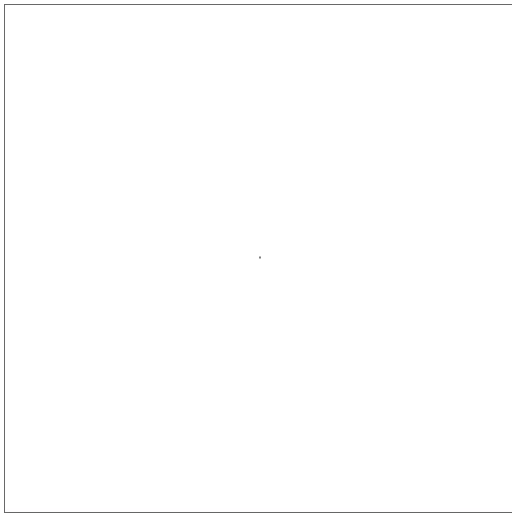
```

procedure help;
begin
  writeln(chr(12), 'Type:');
  writeln(' a plot a point');
  writeln(' b clear the Cellular Automata boundary');
  writeln(' c clear the screen');
  writeln(' d delay is set from the parameter');
  writeln(' h for half step, north-south, then east-west');
  writeln(' i for step up');
  writeln(' j for step left');
  writeln(' l for step right');
  writeln(' m for step down and left');
  writeln(' n for Fredkin's Pattern');
  writeln(' o for step up and right');
  writeln(' p for EF1');
  writeln(' q to quit');
  writeln(' s to step');
  writeln(' u to step up and left');
  writeln(' x for setting px');
  writeln(' y for setting py');
  writeln(' z for setting pz');
  writeln(' , for step down');
  writeln(' . for step down and right');
  writeln(' - for changing the sign of the parameter');
  writeln(' ');
  writeln('Those marked with an asterisk require a numeric parameter,');
  writeln('which should be typed before the letter which runs the');
  writeln('program. For example');
  writeln(' 17e');
  writeln('will run munching squares with 17 for a parameter. ');
  writeln('If you don't give a number, the last number you typed');
  writeln('will be used. ');
  while not keyhit do
end;

```

Apparently you press “b” to “clear the Cellular [sic] Automata boundary”, “n” for “Fredkin’s Pattern” and “p” for “EF1”. And at the end there’s a reference to munching squares. The first pattern above is what you get by pressing “n”; the second by pressing “p”.

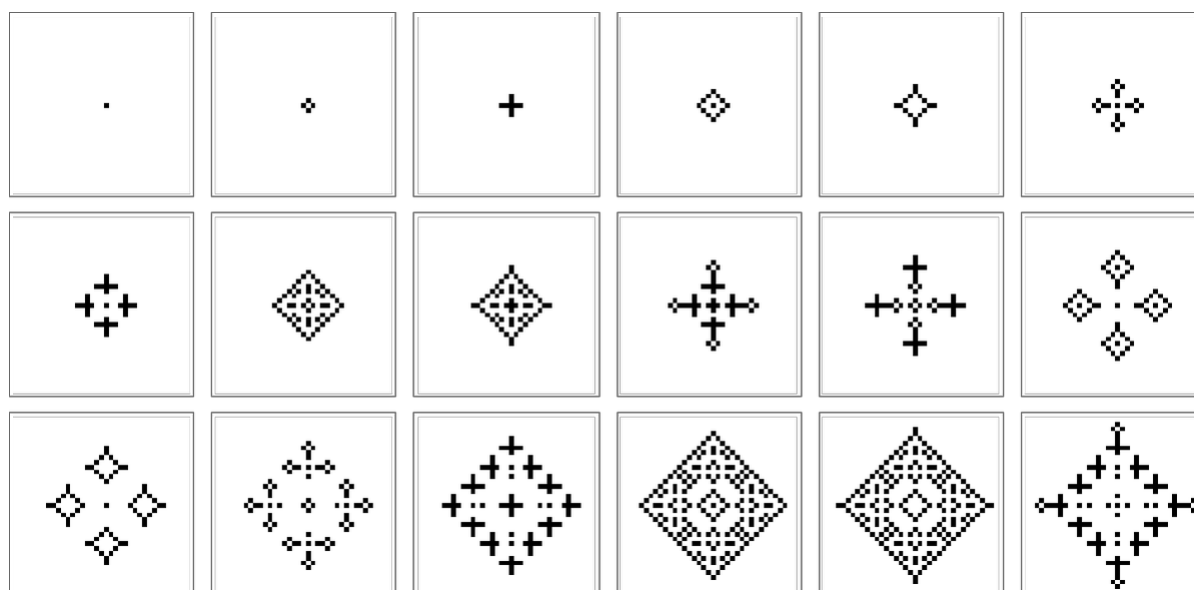
Both patterns look pretty messy. But if instead you press “a”, you get something with a lot more structure:



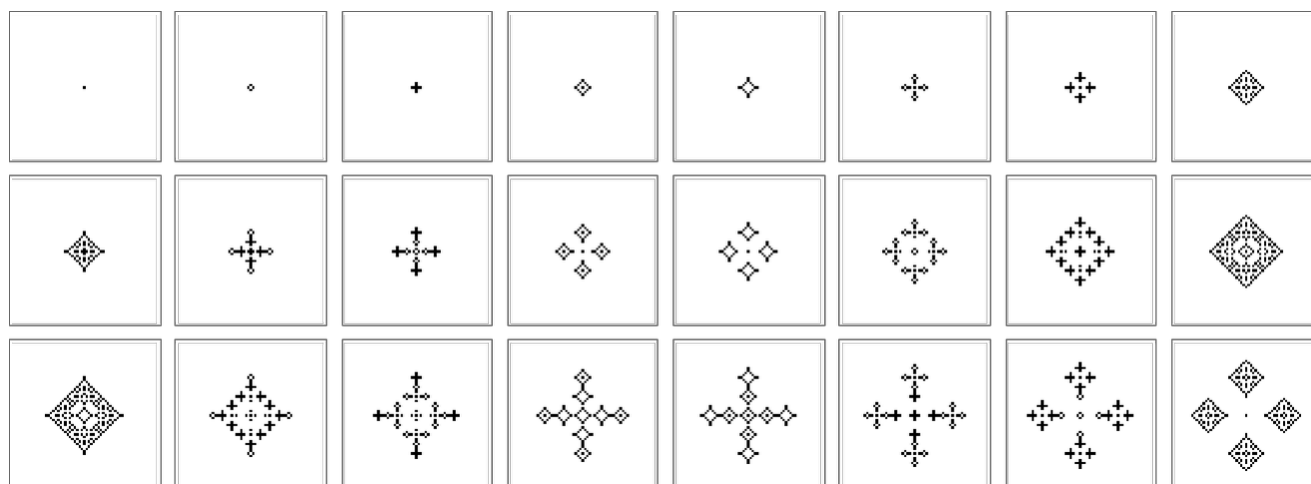
I think Ed showed this to me in passing. But he was more interested in the more complicated patterns, and in the fact that you could get them to reverse what they were doing. And in this animated form, I suspect this just looked to me like another munching squares kind of thing.

But, OK, given that we have the program, can we tell what it actually does? The core of it is a bunch of calls to the function `rasterop( )`. Functions like `rasterop( )` were common in computers with bitmapped displays. Their purpose was to apply a certain Boolean operation to the array of black and white pixels in a region of the screen. Here it’s always `rasterop(6, ...)` which means that the function being applied is Boolean function 6, or Xor (or “sum mod 2”).

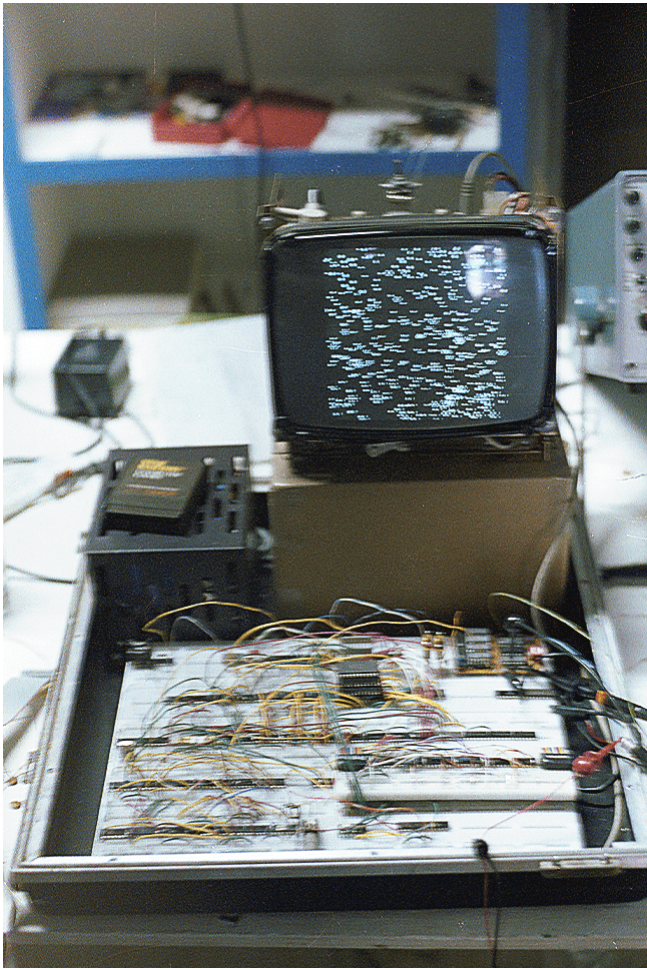
And what’s happening is that chunks of the screen are getting Xor’ed together: specifically, chunks that are offset by one pixel in each of the four directions. And this is all happening in two phases, swapping between different halves of the framebuffer. Here are the central parts of the sequence of frames that get generated starting from a single cell:



It helps a lot to see the separate frames explicitly. And, yes, it’s a cellular automaton. In fact, it’s exactly the “reversible mod-2 rule”. Here it is for a few more steps, with its simple “self-reproduction” increasingly evident:

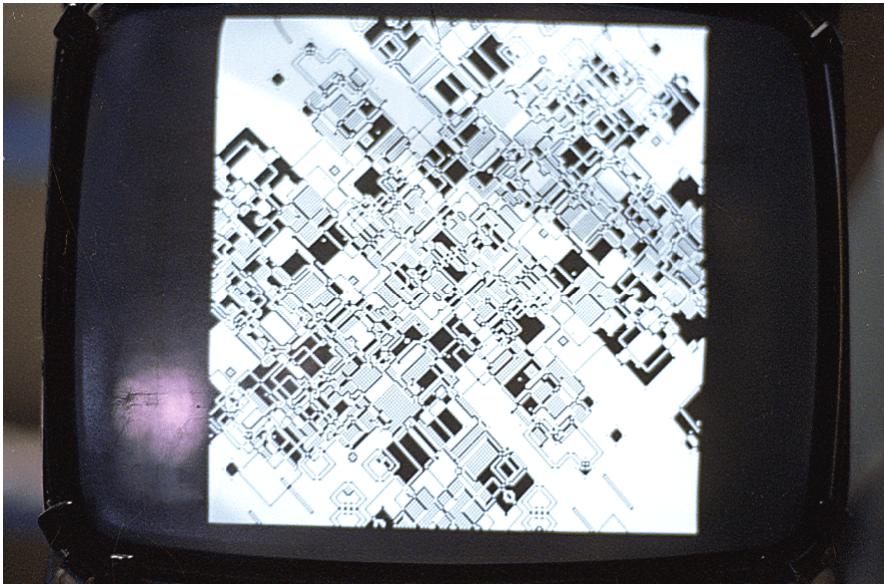


Back in 1982 I think I only saw the PERQ that one time. But in one of the resort cabins on the other side of the island—there was this (as captured in a slightly blurry photograph that I took):



It was a “cellular automaton machine” built out of “raw electronics” by Tom Toffoli and Norm Margolus—who were the core of Ed’s “Information Mechanics” group at MIT. It didn’t feel much like science, but more like a video DJ performance. Patterns flashing and dancing on the screen. Constant rewiring to produce new effects. I wanted to slow it all down and “sciencify” it. But Tom and Norm always wanted to show yet another strange thing they’d found.

Looking in my archives today, I find just one other photograph I took of the machine. I think I considered this the most striking pattern I saw the machine produce. And, yes, presumably it’s a 2D cellular automaton—though despite my decades of experience with cellular automata I don’t today immediately recognize it:



What did I make of Ed back in 1982? Remember, those were days long before the web, and before one could readily look up people’s backgrounds. So pretty much all I knew was that Ed was connected to MIT, and that he owned the island. And I had the impression that he was some kind of technology magnate (and, yes, the island and the plane helped). But it was all quite mysterious. Ed didn’t engage much in technical conversations. He would make statements that were more like pronouncements—that sounded interesting, but were too vague and general for me to do much more than make up my own interpretations for them. Sometimes I would try to ask for clarification, but the response was usually not an explanation, but instead a tangentially related—though often rather engaging—story.

All these years later, though, one particular exchange stands out in my memory. It was at the end of the conference. We were standing around in the little restaurant on the island, waiting for a boat to arrive. And Ed said out of the blue: “I’ll make a deal with you. You teach me how to write a paper and I’ll teach you how to build a company.” At the time, this struck me as quite odd. After all, writing papers seemed easy to me, and I assumed Ed was doing it if he wanted to. And I’d already successfully started a company the previous year, and didn’t think I particularly needed help with it. (Though, yes, I made plenty of mistakes with that company.) But that one comment from Ed somehow for years cemented my view of him as a business tycoon who didn’t quite “get” science, though had ideas about it and wanted to dabble in it.

Ed and Feynman

Ed would later describe Richard Feynman as his best friend. As we discussed above, they'd first met in 1961, and in 1974 Ed had spent the year at Caltech visiting Feynman, having, as Ed tells it, made a deal (analogous to the one he later proposed to me) that he would teach Feynman about computers, and Feynman would teach him about physics. I myself first got to know Feynman in 1978, and interacted extensively with him not only about physics, but also about symbolic computing—and cellular automata. And in retrospect I have to say I'm quite surprised that he mentioned Ed to me only a few times in passing, and never in detail.

But I think the point was that Feynman and Ed were—more than anything else—personal friends. Feynman tended to find “traditional academics” quite dull, and much preferred to hang out with more “unusual” people—like Ed. Quite often the people Feynman hung out with had quite kooky ideas about things, and I think he was always a little embarrassed by this, even though he often seemed to find it fun to indulge and explore those ideas.

Feynman always liked solving problems, and applying himself to different kinds of areas. But I have to say that even I was a little surprised when in writing this piece I was going through [the archives of Ed's papers at MIT](#), and found the following letter from Feynman to Ed:

CALIFORNIA INSTITUTE OF TECHNOLOGY

CHARLES C. LAURITSEN LABORATORY OF HIGH ENERGY PHYSICS
PASADENA, CALIFORNIA 91125

June 17, 1976

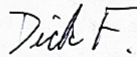
Mr. E. Fredkin
Computer Sciences
Massachusetts Institute of Technology
Cambridge, MA 02139

Dear Fredkin:

We enjoyed your visit very much. You'll have to come back here more often (of course its impossible for me to ever go to Boston).

I am enclosing a description of the cutting machine which we can't sell. Anything you can do to help or make some suggestion as to how we can do something with it would be wonderful.

Sincerely,



Richard P. Feynman

RPF;ht

Encl.

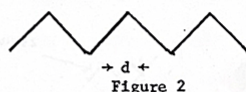
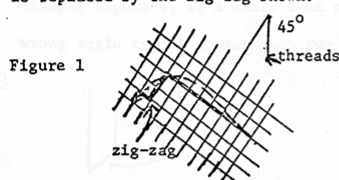
Clearly he—like me—viewed Ed as an authority on business. But what on earth was this “cutting machine”, and why was Feynman trying to sell it?

For what it's worth, the next couple of pages tell the story:

R. P. Feynman

Description of Principle of the Estephanian Cutting Machine

The usual way to cut cloth is to move a vertical oscillating knife through a stack of layers of cloth, the knife following the curves of the desired pattern or marker. It has proved difficult to mechanize this process because of the adjustments of the blades angle to the direction of the width (transverse direction) as it moves both transversely (across the width) and longitudinally (along the length) of the cloth. The invention described here results from the observation that for most applications it is sufficient (and indeed, often advantageous) if the cut edge is not a smooth curve but is instead made up of little sequences of straight lines of fixed direction as in pinking. We take these directions to be $\pm 45^\circ$ to the longitudinal direction and hence to the direction of the threads. Thus the curved line to be cut, in figure 1, is replaced by the zig-zag shown.



Such a cut is easily made mechanically. First imagine a transverse row of knives alternately $\pm 45^\circ$ to the longitudinal direction across the entire cloth width (see figure 2). This is for explanation only, (in a moment we shall describe a more practical mechanical arrangement). The repARATION of the knives we call d . (For example, for $d = .09$ inches there will be 800 knives in a 72" width of cloth and the length of each little cut will be $\sqrt{2} d$ or close to $1/8$ ".)

These knives are plunging knives set above the cloth as seen in figure 3 where we look at one knife from the side. The knives are to be independently

-2-

The cutting machine proper (not including the machine which prepares the markers, nor the equipment for setting up the layers of material to feed into the machine) might be about 15 feet long by 8 feet wide, table height, weigh about 3 tons, take about 2 or 3 kilowatts of electrical power. We have made no careful cost estimate and can only guess by comparison with other similar machines. Judging this way we think it may cost in the neighborhood of \$50,000, (not including development costs).

The machine must be developed, a complete prototype does not exist. However no isoteric new principles are involved in its operation, it is based on state-of-the-art technology and can be developed using only routine engineering procedures. There are a large number of parallel elements all acting in the same way. We have tested what could be considered a prototype of one of these elements. We have checked the critical questions. We have found that we can easily control the action of the element at the rate of repetition we need. In addition we have made cuts at this rate with this device on a wide range of fabrics and have verified that in every case the cloth is cut cleanly and rapidly. We know of no point at which the machine is not well within the range of conditions where it should not easily operate as planned. We have not, however, completed any tests of machine wear or cutting blade life.

The machine is U.S. patented. Application has also been made in a number of foreign countries.

The principle inventor is Steve Estephanian. A co-developer is R. P. Feynman, Professor of Physics at the California Institute of Technology. He is a friend and neighbor of Mr. Estephanian and has collaborated with him on other inventions.

and neighbor of Mr. Stepanian and has collaborated with him on other inventions and developments.

The services of both these inventors would be available under suitable arrangements to assist any persons attempting the development of this machine.

Feynman's next-door neighbor had a company that made swimwear, and this was a machine for cutting the necessary fabric—and Feynman had helped develop it. And much as Feynman had been prepared to help his neighbor with this, he was also prepared to help Ed with some of his ideas about physics. And in the archive of Ed's papers, there's a letter from Feynman:

CALIFORNIA INSTITUTE OF TECHNOLOGY

CHARLES C. LAURITSEN LABORATORY OF HIGH ENERGY PHYSICS
PASADENA, CALIFORNIA 91125

February 26, 1980

Mr. E. Fredkin
Artificial Intelligence Group
Massachusetts Institute of Technology
Cambridge, MA 02139

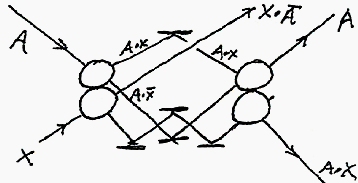
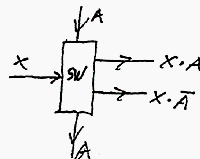
Dear Ed:

I have amused myself thinking of ball models.

The one you left on my board had some minor errors. The $\bar{A}$ line intersected the $A \cdot B$ line and also the $A \cdot C$ line. Also the 1 line containing the recirculating ball intersects the output line $A \rightarrow C, B$ and the input line C .

Just for fun I tried to make one without a circulating ball at all. I ~~shall~~ send you a copy of it.

I also noticed that the following gate that can be called a switch (for x is switched onto one output line or the other depending on whether $A=1$ or 0 - the control line comes through) (it is a Fredkin gate with one entry wire 0) is easy to make with a ball model:

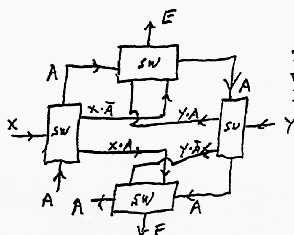
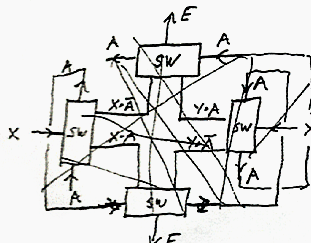


The switch is, of course, reversible provided it is never fed with two ones backwards (which rule makes it easy to make mistakes in circuits). It is

easy to make a Fredkin gate if $A=1$ and E, F if $A=0$ out of switches as follows:

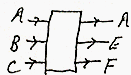


where A switches X, Y into F, E if $A=1$ and E, F if $A=0$ out of switches as follows:



That is how I know we don't need a recirculating ball.

FREDKIN GATE

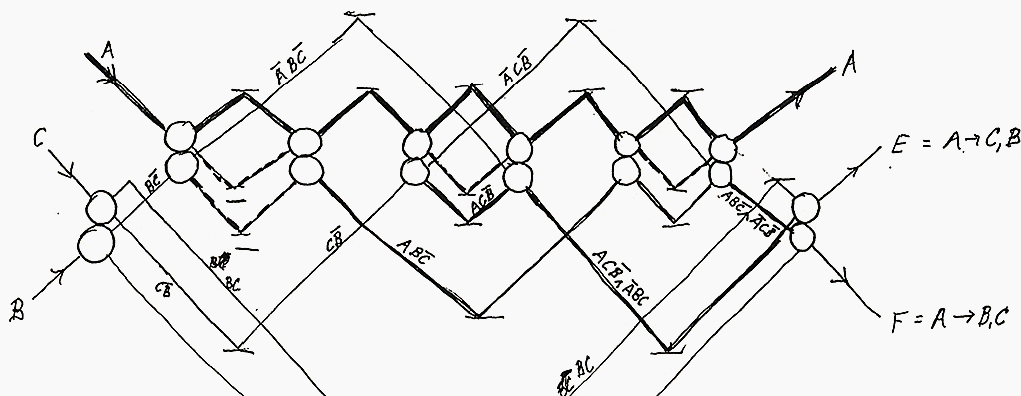


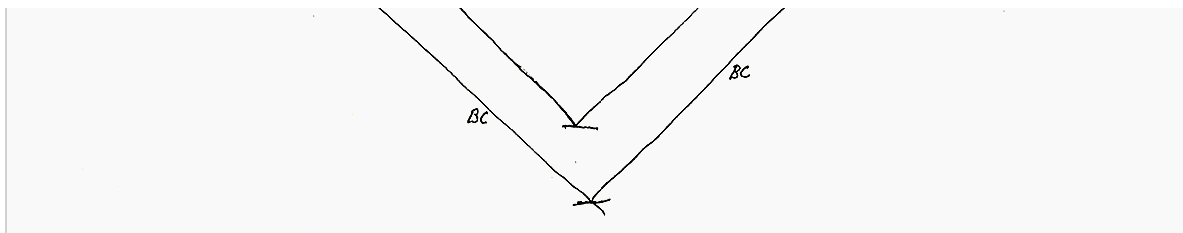
$E, F = B, C$ if $A=0$
 $E, F = C, B$ if $A=1$

CHANNELS USED IF $A=1$
MARKED BY HEAVY LINES

By FREDKIN BALLS.

No RECYCLING BALLS.





I don't know whether this is the first place the term "Fredkin gate" was ever used. But what's here is a quintessential example of Feynman diving into some new subject, doing detailed calculations (by hand) and getting a useful answer—in this case about what would become Ed's best-known invention: reversible logic, and the Fredkin gate.

Feynman had always been interested in "computing". And indeed when he was recruited to the Manhattan Project it was to run a team of human computers (equipped with mechanical desk calculators). I think Feynman always hoped that physics would "become computational" at least in some sense—and he would for example lament to me that Feynman diagrams were such a bad way to compute things. Feynman always liked the methodology of traditional continuous mathematics, but (as I just noticed) even in 1964 he was saying that "I believe that the theory that space is continuous is wrong, because we get these infinities and other difficulties...". And elsewhere in [his 1964 lectures](#) that became *The Character of Physical Law* Feynman says:

It always bothers me that, according to the laws as we understand them today, it takes a computing machine an infinite number of logical operations to figure out what goes on in no matter how tiny a region of space, and no matter how tiny a region of time. How can all that be going on in that tiny space? Why should it take an infinite amount of logic to figure out what one tiny piece of space/time is going to do? So I have often made the hypothesis that

ultimately physics will not require a mathematical statement, that in the end the machinery will be revealed, and the laws will turn out to be simple, like the chequer board with all its apparent complexities. But this speculation is of the same nature as those other people make – 'I like it', 'I don't like it', – and it is not good to be too prejudiced about these things.

Did Feynman say these things because of his conversations with Ed? I rather doubt it. But as I was writing this piece I learned that Ed thought differently. [As he told it:](#)

I never pressed any issue that would sort of give me credit, okay? It's just my nature. A very weird thing happened toward the end of my time at Caltech. Richard Feynman and I would get into very fierce arguments. . . . I'm trying to convince him of my ideas, that at the bottom is something finite and so on. He suddenly says to me, "You know, I'm sure I had this same idea sometime quite a while ago, but I don't remember where or how or whether I ever wrote it down." I said, "I know what you're talking about. It's a set of lectures you gave someplace. In those lectures you said perhaps the world is finite." He just has this little statement in this book. I saw the book on his shelf. I got it out, and he was so happy to see that there. What I didn't tell him was he gave that lecture years after I'd been haranguing him on this subject. I knew he thought it was his idea, and I left it that way. That was just my nature.

Notwithstanding what he said, I rather suspect he did push the point. And for example when Feynman gave a talk on "[Simulating Physics with Computers](#)" at the [1981 MIT Physics of Computation conference](#) that Ed co-organized, he was careful to write that:

talk about. I want to talk about the problem of simulating physics with computers and I mean that in a specific way which I am going to explain. The reason for doing this is something that I learned about from Ed Fredkin, and my entire interest in the subject has been inspired by him. It

Ed, by the way, arranged for Feynman to get his first personal computer: a Commodore PET. I don't think Feynman ended up using it terribly much, though in 1984 he took it with him on a trip to Hawaii where he and his son Carl used it to work out probabilities to try to "[crack](#)" the randomness of my rule 30 cellular automaton (needless to say, without success).

Digital Physics & Reversible Logic

Back at MIT in 1975 after his year at Caltech, Ed was no longer the director of Project MAC, but was still on the books as a professor, albeit something of an outcast one. Soon, though, he was teaching a class about his ideas—under the title of "Digital Physics":

DIGITAL PHYSICS 6.895

Digital Physics is a graduate subject for both computer science and physics students. 6.895 is about relations between concepts of physics and concepts of computer science. Physicists must occasionally be computer users and computer scientists are aware of the results of solid state physics, but so far there has been very little intellectual communication between the two fields. Digital Physics is an attempt to bridge that gap by presenting to students from both fields results obtained by applying the ideas from one field to some of the most basic open questions of the other field. What has emerged is a new way of looking at both areas, and a new area of research called Digital Physics. Underlying the research are speculations about modelling some microscopic physical events more accurately by digital computation than by the mathematics of continuous variables and that computation might be a basic phenomenon that is governed by the laws of physics.

So far, some of the more startling results of research in Digital Physics include the discovery that there is no theoretical minimum to the energy that must be dissipated by a computation. This means that a computer might be built that uses no energy except when communicating. Another result is that there are simple digital versions of fundamental equations of physics, such as the Schrodinger equation or Newton's laws, that can operate without ever losing any information or accumulating any long term error. This is surprising because during a computation there are small errors (roundoff and truncation) and information seems to disappear, yet the overall computation conserves information.

As is indicated by the course description and the lecture outline, a great deal of material will be covered. There will be some reading, in the Feynman Lectures, Volume III, in Computation, Finite and Infinite Machines by Minsky and in the lecture notes. There will be a few problem sets and each student will be required to write one paper. 6.895 will meet on Tuesday, Thursday and Friday at 2:00 PM in 36-153

Cellular automata weren't specifically mentioned in the course description—though in the syllabus they were there, with the Game of Life as a key example:

Digital Physics

Physicists job is to understand the functioning of the universe.

Laws, such as Newton's or Maxwell's, vs. entities, such as mass, electrons, etc.

Newton's laws were right, his entities were wrong: point mass particles, infinite propagation speed forces.

Possible entities are principally limited by current mathematical technique.

The three principal entities in physics: partical, field, wave. Corresponding math: ordinary calculus; vector calculus; differential equations and transforms. Other examples: curved space -- Riemannian geometry and tensor calculus.

Digital physics is based on the mathematics of computation and information instead of the various branches of calculus.

Why would we want a different math for physics? QM is correct on an atomic scale, but as yet unproven on the scale of "fundamental particles". Many mysteries, things unexplained. But QM is so sucessful, its got to be right. This is a fundamental philosophical question on scientific method. Newtonian mechanics as counterexample. Math as "universal computer".

Basics of DP: Physical law specifies a computation, entities are information in the strict sense, data structures. The principal of finite information content. The abandonment of continuous variables.

Status of DP: Much interest, but no theories that could withstand experimental verification. Very refreshing, compelling way of thinking. New possibilities, such as roundoff error. Easier to understand: space, time, matter, the speed of light, the beginning. Harder to understand: inertia, relativity, isotropic.

The cellular automaton. Definition: space, matter, time, physical law. Nondeterministic CA. Reversible CA, the Garden of Eden theorem.

Some history of CA.

Life. The physics. The chemistry. Static forms, oscillators, motion.

Trying to explain classical mechanics with CA. Why? Hope that QM will fall out, but that's not likely. Easier to understand. World behaves classical on large scale. Fundamental difficulties need to be resolved.

The difficulties: uniform motion; relativity; isotropic properties; independence of coordinates.

Uniform motion. A solution in one dimension. Unary representations in CA. Interactions: fields, force, acceleration.

The 2-D problem. Binary wave mechanics. Curved space.

Size of quantum of space and time. The Plank length, time, and mass.

Back in the 1960s, cellular automata had been a popular topic in theoretical computer science. But by the mid-1970s the emphasis of the field had switched to things like computational complexity theory—and, as Ed told me many times, his efforts to interest people at MIT in cellular automata failed, with influential CS professor Albert Meyer (whose advisor Patrick Fischer had worked quite extensively on cellular automata) apparently telling Ed that “one can tell someone is out of it if they don’t think cellular automata are dead”. (It’s an amusing irony that around this time, Meyer’s future wife Irene Greif would point John Moussouris—who we’ll meet later—to Ed and his work on cellular automata.)

Ed’s ideas about physics were not well received by the physicists at MIT. And for example when students from Ed’s class asked the well-known MIT physics professor Philip Morrison what he thought of Ed’s approach, he apparently responded that “Of course Fredkin thinks

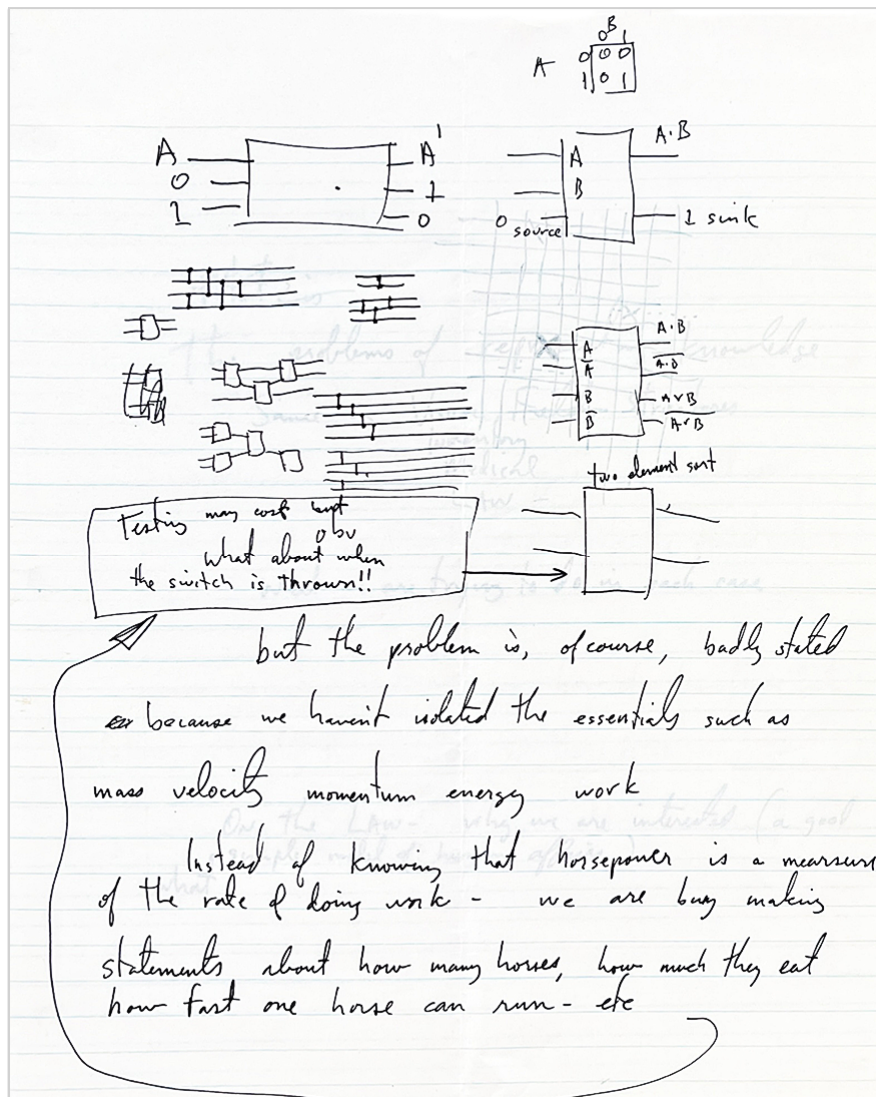
the universe is a computer—he’s a computer person; if instead he were a cheese merchant he’d think it was a big cheese!”

When Ed was at Caltech in 1974 a big focus there—led by Carver Mead—was VLSI design. And this led to increasing interest in the ultimate limits on computation imposed by physics. Ever since von Neumann in the 1950s it had been assumed that every step in a computation would necessarily require dissipation of energy—and this was something Carver Mead took as a given. But if this was true, how could Ed’s cellular automaton for the universe work? Somehow, Ed reasoned, it—and any computation, for that matter—had to be able to run reversibly, without dissipating any energy. And this is what led Ed to his most notable scientific contribution: the idea of [reversible logic](#).

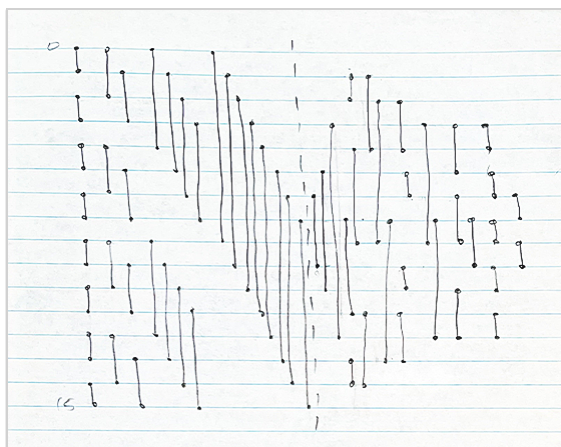
Ordinary logic operations—like And and Or—take two bits of input and give one bit of output. And this means they can’t be reversible: with only one bit in the output there isn’t information to uniquely determine the two bits of input from the output. But if—like Ed—you consider a generalized logic operation that for example has both two inputs and two outputs, then this can be invertible, i.e. reversible.

The concept of an invertible mapping had long existed in mathematics, and under the name “automorphisms of the shift” had even been studied back in the 1950s for the case of what amounted to 1D cellular automata (for applications in cryptography). And in 1973 Charles Bennett had shown that one could make a reversible analog of a Turing machine. But what Ed realized is that it’s possible to make something like a typical computer design—and have it be reversible, by building it out of reversible logic elements.

Looking through the archive of Ed’s papers at MIT, I found what seem to be notes on the beginning of this idea:



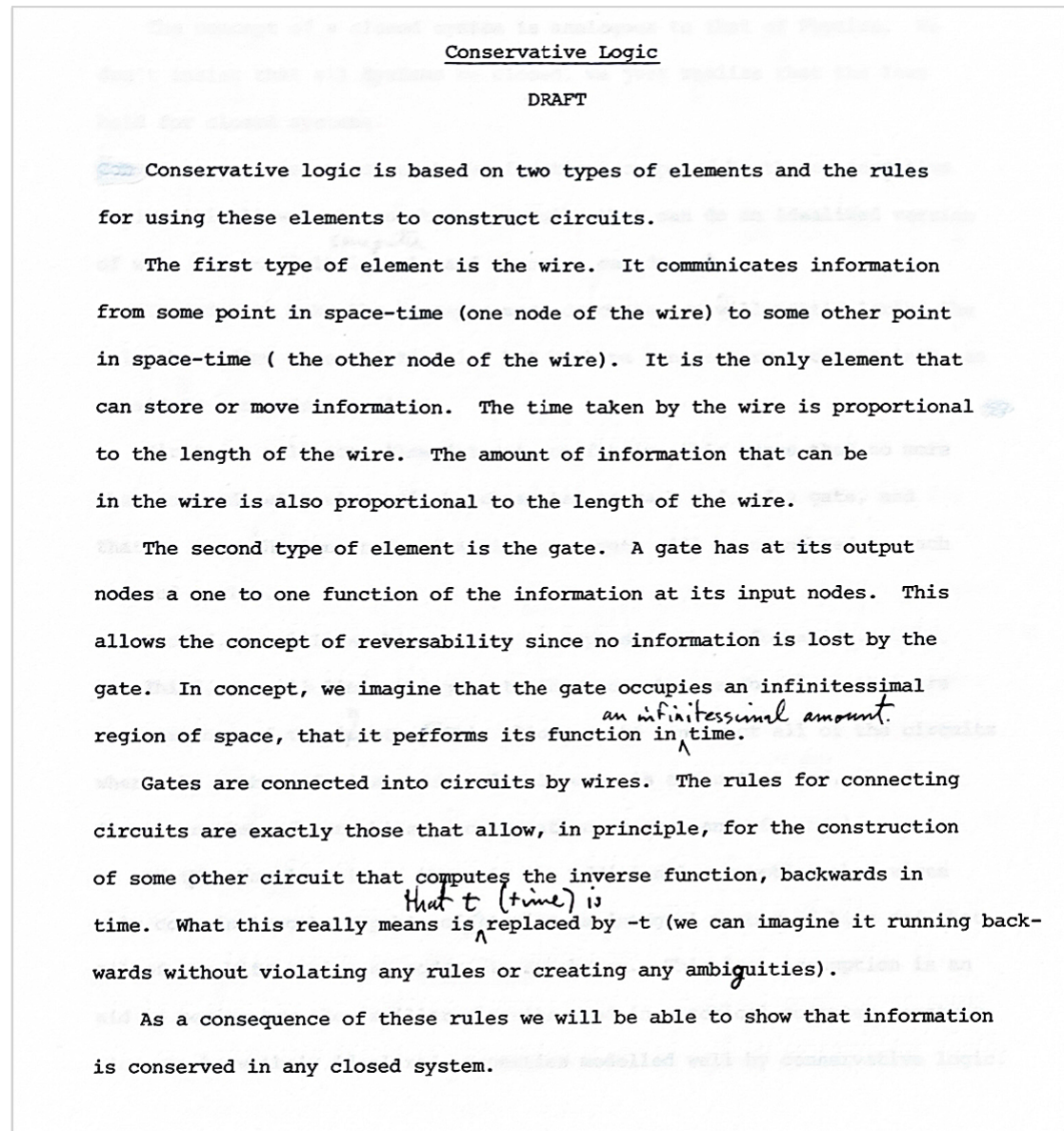
And I also found this—which I immediately recognized as a [sorting network](#), in which values get sorted through a sequence of binary comparisons:



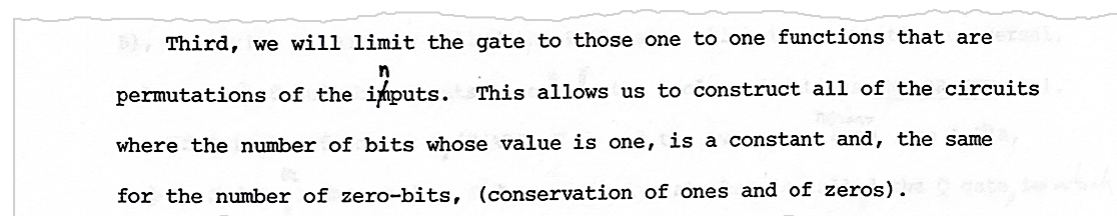
Sorting networks are inevitably reversible. And this particular sorting network I recognized as the largest guaranteed-optimal sorting network that's known—discovered by Milton

Green at SRI (then “Stanford Research Institute”) in 1969. It’s implausible that Ed independently discovered this exact same network, but it’s interesting that he was drawing it (by hand) on a piece of paper.

Ed’s archives also contain a 3-page draft entitled “Conservative Logic”:



Ed explains that he is limiting himself to gates that implement permutations



and then goes on to construct a “symmetric-majority-parity” gate—which he claims is “computation universal”:

Consider the following class of gates with six binary nodes (three input nodes, A, B & C; three output nodes, D, E & F.)

A	D
B	E
C	F

The rule for this class of gate is that a binary function, $F(A, B, C)$ determines which of two permutations of the inputs will be presented to the outputs.

For example if F is the parity function (sum modulo 2) and the permutations are the two possible rotations ($A \Rightarrow B, B \Rightarrow C, C \Rightarrow A$ or $A \Rightarrow C, B \Rightarrow A, C \Rightarrow B$), we arrive at what is called the SMP* gate, which is computation universal, (by means of suitable inputs it can do the boolean functions AND OR NOT etc).

If F is the function A , ($F(ABC) = A$) and the two permutations are $A \Rightarrow A, B \Rightarrow B, C \Rightarrow C$ or $A \Rightarrow A, B \Rightarrow C, C \Rightarrow B$, we arrive at what is called the Q gate, which is also computation universal.

*SMP stands for symmetric-majority-parity gate - which will be explained later.

It's not quite a Fredkin gate, but it's close. And, by the way, it's worth pointing out that these gates alone aren't “computation universal” in something like the Turing sense. Rather, the point is that—like with Nand for ordinary logic—any reversible logic operation (i.e. permutation) with any number of inputs can be constructed using just these gates, connected by wires.

Ed didn't at first publish anything about his reversible logic idea, though he talked about it in his class, and in 1978 there were already students writing term papers about it. But then in 1978, as [Ed told it later](#):

I found this guy Tommaso Toffoli. He had written a paper that showed how you could build a reversible computer by storing everything that an ordinary computer would have to forget. I had figured out how to have a reversible computer that didn't store anything because all the fundamental activity was reversible. Okay? So I decided to hire him because he was the only person who tried to do it and he didn't succeed, really, and I had—and I hired him to help me.

Toffoli had done a first PhD in Italy building electronics for cosmic ray detectors, and in 1978 he'd just finished a second PhD, working on 2D cellular automata with Art Burks (who

had coined the name “cellular automaton”). Ed brought Toffoli to MIT under a grant to build a cellular automaton machine—leading to the machine I saw on Ed’s island in 1982. But Ed also worked with Toffoli to write a paper about conservative logic—which finally appeared in 1982, and contained both the Fredkin gate, and the Toffoli gate. (Ed later griped to me that Toffoli “really hadn’t done much” for the paper—and that after all the Toffoli gate was just a special case of the Fredkin gate.)

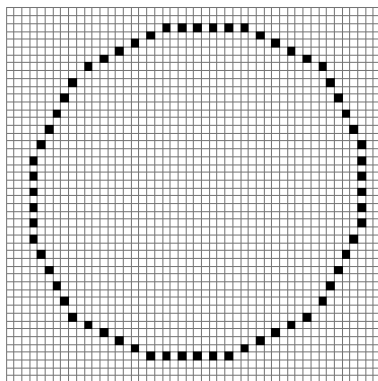
Back in 1980—on the way to this paper—Ed, with Feynman’s encouragement, had had another idea: to imagine implementing reversible logic not just abstractly, but through an explicit physical process, namely collisions between elastic billiard balls. And as we saw above, Feynman quickly got into analyzing this, for example seeing how a Fredkin gate could be implemented just with billiard balls.

But ultimately Ed wanted to implement reversibility not just for things like circuits, but also—imitating the reversibility that he believed was fundamental to physics—for cellular automata. Now the fact is that reversibility for cellular automata had actually been quite well studied since the 1950s. But I don’t think Ed knew that—and so he invented his own way to “get reversibility” in cellular automata.

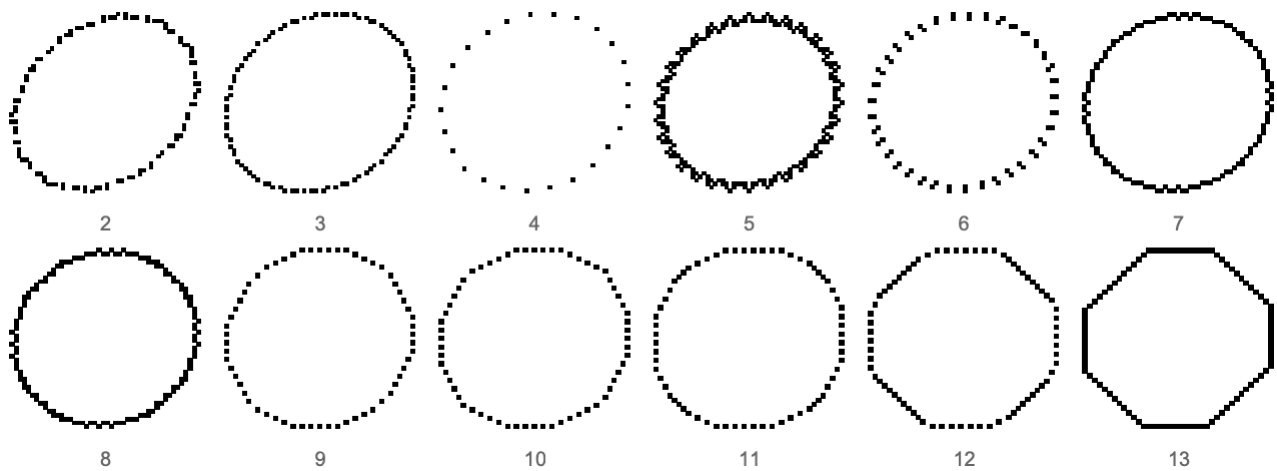
It came from something Ed had seen on the PDP-1 back in 1961. As Ed tells it, in playing around with the PDP-1 he had come up with a piece of code that surprised him by drawing something close to a circle in pixels on the screen. Minsky had apparently “gone into the debugger” to see how it worked—and in 1972 HAKMEM attributed the algorithm to Minsky (though in the Pascal program I got from Ed in 1982, it appears as a function called `efpattern( )`). Here’s a version of the algorithm:

```
f[ d_ ] [ { x_, y_ } ] := With [ { z = x - Quotient [ y, d ] }, { z, y + Quotient [ z, d ] } ]
```

```
Graphics [ Point [ NestList [ f [ 10 ], { 0, 25 }, 100 ] ] ]
```



And, yes, with different divisors d it can give [rather different](#) (and [sometimes wild](#)) results:



But for our purposes here what’s important is that Ed found out that this algorithm is reversible—and he realized that in some sense the reason is that it’s based on a second-order recurrence. And, once again, the basic ideas here are well known in math (cf. reversibility of the wave equation, which is second order). But Ed had a more computational version: a [second-order cellular automaton](#) in which one adds mod 2 the value of a cell two steps back. And I think in 1982 Ed was already talking about this “mod-2 trick”—and perhaps the PERQ program was intended to implement it (though it didn’t).

Ed’s work on reversible logic and “digital physics” in a sense came to a climax with the [1981 Physics of Computation conference at MIT](#)—that brought in quite a *Who’s Who* of people who’d been interested in related topics (as I mentioned above, I wasn’t there because of a clash with the release of SMP Version 1.0, though I did meet or at least correspond with most of the attendees at one time or another):



Physics of Computation Conference Endicott House MIT May 6-8, 1981

- | | | | |
|---------------------|----------------------|-------------------|--------------------|
| 1 Freeman Dyson | 13 Frederick Kantor | 25 Robert Suaya | 37 George Michaels |
| 2 Gregory Chaitin | 14 David Levin | 26 Stan Kugell | 38 Richard Feynman |
| 3 James Crutchfield | 15 Konrad Zuse | 27 Bill Gosper | 39 Laurie Lingham |
| 4 Norman Packard | 16 Bernard Zeigler | 28 Lutz Priese | 40 Thiagarajan |
| 5 Panos Ligomenides | 17 Carl Adam Petri | 29 Madhu Gupta | 41 ? |
| 6 Jerome Rothstein | 18 Anatol Holt | 30 Paul Benioff | 42 Gerard Vichniac |
| 7 Carl Hewitt | 19 Roland Vollmar | 31 Hans Moravec | 43 Leonid Levin |
| 8 Norman Hardy | 20 Hans Bremerman | 32 Ian Richards | 44 Lev Levitin |
| 9 Edward Fredkin | 21 Donald Greenspan | 33 Marian Pour-El | 45 Peter Gacs |
| 10 Tom Toffoli | 22 Markus Bueettiker | 34 Danny Hillis | 46 Dan Greenberger |
| 11 Rolf Landauer | 23 Otto Floberth | 35 Arthur Burks | |
| 12 John Wheeler | 24 Robert Lewis | 36 John Cocke | |

Originally Ed wanted to call the conference “Physics and Computation”. But Feynman objected, and the conference was renamed. In the end, though, Feynman gave a talk entitled “Simulating Physics with Computers”—which most notably talked about the relation between quantum mechanics and computation, and is often seen as a key impetus for the development of quantum computing. (As a small footnote to history, I worked with Feynman quite a bit on the possibility of both quantum computing and quantum randomness generation, and I think we were both convinced that the process of measurement was ultimately going to get in the way—something that with our [Physics Project](#) we are finally now beginning to be able to [analyze in much more detail](#).)

But despite his interactions with Feynman, Ed was never too much into the usual ideas of quantum mechanics, hoping (as he said in the flyer for his course on digital physics) that perhaps quantum mechanics would somehow fall out of a classical cellular-automaton-based universe. But when quantum computing finally became popular in the 1990s, reversible logic was a necessary feature, and the Fredkin gate (also known as CSWAP or “controlled-swap”) became famous. (The Toffoli gate—or CCNOT—is a bit more famous, though.)

In tracing the development of Ed’s ideas, particularly about “digital physics”, there’s another event worthy of mention. In late 1969 Ed learned about an older German tech entrepreneur named Konrad Zuse who’d published an article in 1967 (and a book in 1969) on *Rechnender Raum (Calculating Space)*—mentioning the term “cellular automata”:

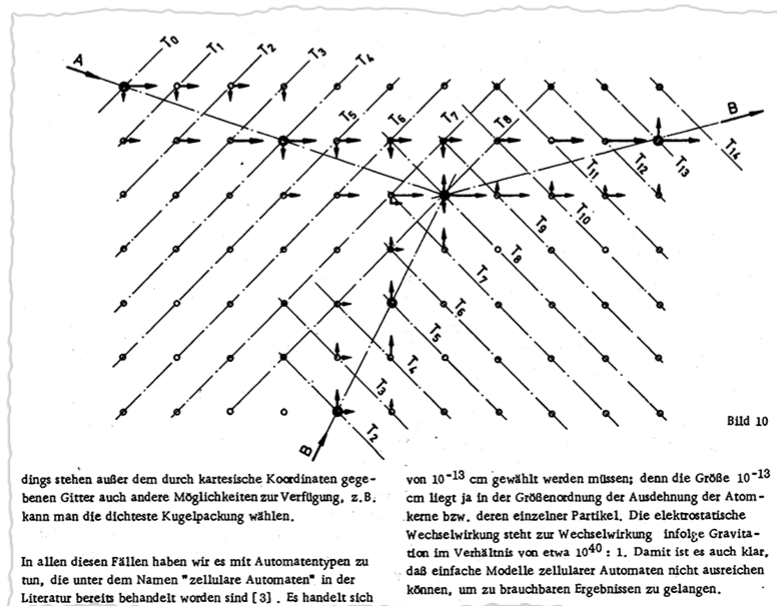
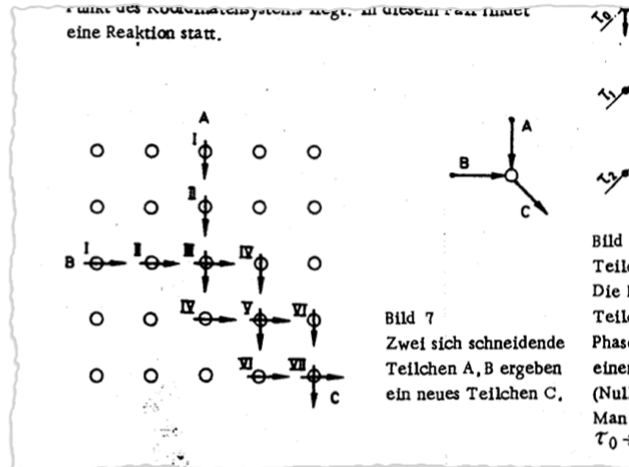
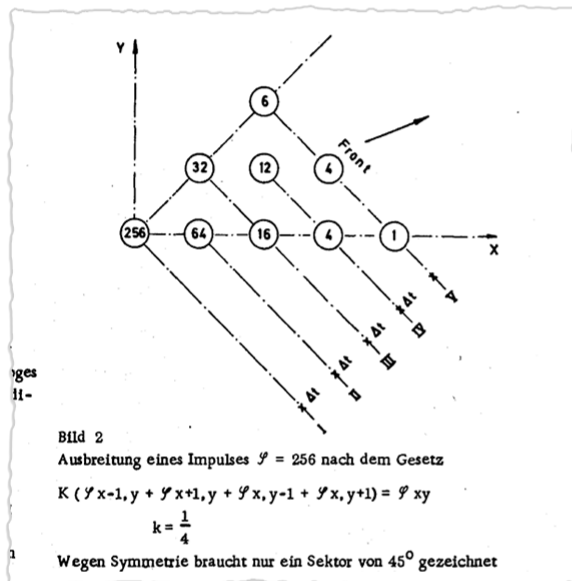
Rechnender Raum

K. Zuse, Bad Hersfeld

Mit 10 Bildern

Zusammenfassung: Der Verfasser unternimmt den Versuch, informations- und automatentheoretisches Denken auf physikalische Probleme anzuwenden. Neben einigen allgemeinen Betrachtungen wird im wesentlichen der Gedanke einer Digitalisierung räumlicher Beziehungen verfolgt, womit die Idee der Quantisierung der physikalischen Größen weiter verallgemeinert wird.

Summary: In the following contribution the author tries to apply an information and automata theory approach to certain problems of physics. Besides some general considerations, the main thought discussed is the idea of digitalizing spatial relations whereby the conception of quantization of the physical quantities is generalized.



Although Zuse was 24 years older than Ed, there were definitely similarities between them. Zuse had been very early to computers, apparently building one during World War II that suffered an air raid (and may yet still lie buried in Berlin). After the war, Zuse started a series of computer companies—and had ideas about many things. He'd been trained as an

engineer, and perhaps it was having worked on solving his share of PDEs using finite differences that led him to the idea—a bit like Ed’s—that space might fundamentally be a discrete grid. But unlike Ed, Zuse for the most part seemed to think that—as with finite differences—the values on the grid should be continuous, or at least integers. Ed arranged for Zuse’s book to be translated into English, and for Zuse to visit MIT. I don’t know how much influence Zuse had on Ed, and when Ed talked to me about Zuse it was mostly just to say that people had treated his ideas—like Ed’s—as rather kooky. (I exchanged letters with Zuse in the 1980s and 1990s; he seemed to find my work on cellular automata interesting.)

Ideas & Inventions Galore

It wasn’t just physics that Ed had ideas about. It was lots of other things too. Sometimes the ideas would turn into businesses; more often they’d just stay as ideas. Ed’s archive, for example, contains a document on the “Intermon Idea” that Ed hoped would “provide a permanent solution to the world’s problem of not having a stable medium of exchange”:

Ed Fredkin's Intermon Idea
Working draft #1
4/22/76

The compelling reason for the creation of the Intermon unit of exchange is to provide a permanent solution to the world's problem of not having a stable medium of exchange.

Intermons will be a medium of exchange, similar to dollars, or pounds, or gold. But where dollars, or pounds, or gold are tied to their respective countries or to a particular market for their inherent value (that is, their value compared to other mediums of exchange), Intermons will be tied to a total commodity index.

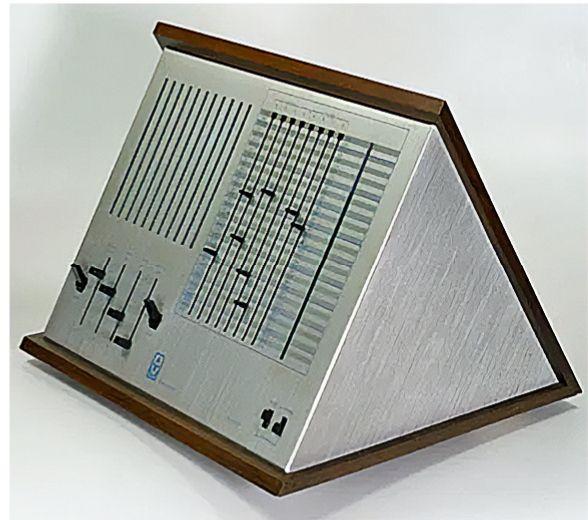
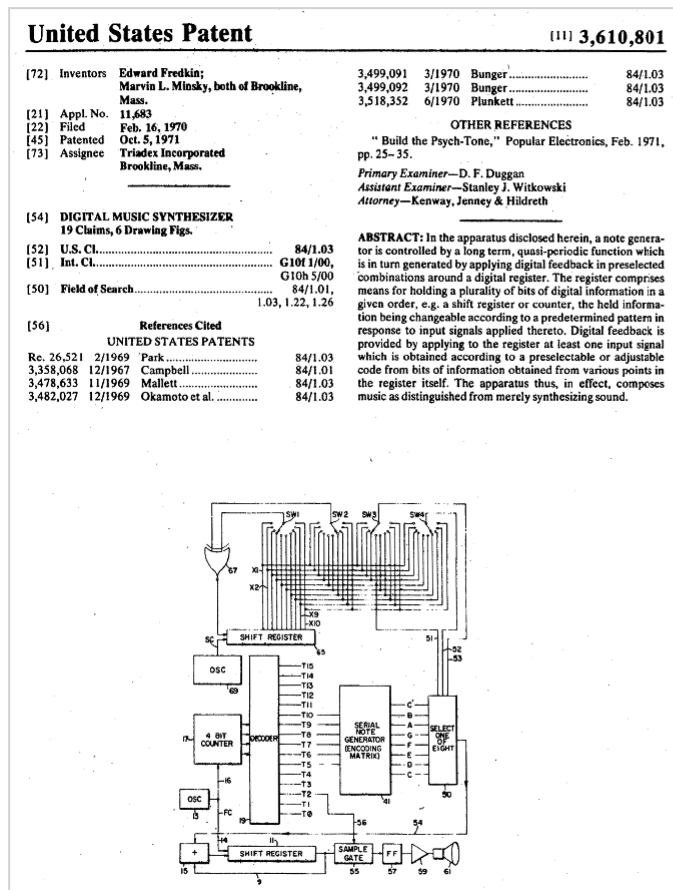
The commodity index will cover all important world commodities, and most, or perhaps all, of the lesser important ones. There will exist some formula, based upon the total value (price times quantity) of the indexing commodities, and this formula will be readily available, so that anyone could calculate the value of an Intermon compared to the value of any other medium of exchange.

The advantage Intermons will have over other currencies is that they will tend to be more stable. This follows from the fact that the inherent value of an Intermon is tied directly to the commodities which ultimately determine the value of all mediums of exchange.

To implement the Intermon concept, an Intermon Bank must be set up. Once in operation, the Bank will loan Intermons and accept deposits of Intermons. Interest will be paid on deposits, and interest will be charged on loans. The gap in interest rates between loans and deposits will remain fixed, and will float on a daily basis. In order to keep deposits and loans in balance, the interest rates will be adjusted daily; it is possible that a negative rate will be paid on deposits. The Bank's profits come from the interest gap. Reserves of the Bank will be held in commodities futures to insure the underlying value and stability of the Intermon.

And, no, Ed wasn't Satoshi Nakamoto—though he did tell me several times that (although, to his displeasure, it was never acknowledged) he had suggested to Ron Rivest (the “R” of RSA cryptography) the idea of “using factoring as a trapdoor”. And—not content with solving the financial problems of the world, or, for that matter, fundamental physics—Ed also had his “algorithmic plan” to prevent the possibility of World War III.

And then there was the Muse. Marvin Minsky had long been involved with music, and had assembled out of electronic modules a system that generated sequences of musical notes. But in 1970 Ed and Minsky developed what they called the Muse—whose idea was to be a streamlined system that would use integrated circuits to “[automatically compose music](#)”:



In actuality, the Muse produced sequences of notes determined by a [linear feedback shift register](#)—in essence a 1D additive cellular automaton—in which the details of the rule were set on its front panel as “themes”. The results were [interesting—if rather R2-D2-like](#)—but weren’t what people usually thought of as “music”. Ed and Minsky started a company named Triadex (note the triangular shape of the Muse), and manufactured a few hundred Muses. But the venture was not a commercial success.

Particularly through interacting with Minsky, Ed was quite involved in “things that should be possible with AI”. The Muse had been about music. But Ed also for example thought about chess—where he wanted to build an array of circuits that could tree out possible moves. Working with Richard Greenblatt (who had developed an earlier chess machine) my longtime friend John Moussouris ended up designing [CHEOPS](#) (a “[Chess-Oriented Processing System](#)”) while Ed was away at Caltech. (Soon thereafter, curiously enough, Moussouris would go to Oxford and work with Roger Penrose on discrete spacetime—in the form of spin networks. Then in later years he would found two important Silicon Valley microprocessor companies.)

Keeping on the chess theme, Ed would in 1980 (through his Fredkin Foundation) put up the Fredkin Prize for the first computer to beat a world champion at chess. The first “pre-prize” of \$5k was awarded in 1981; the second pre-prize of \$10k in 1988—and the grand prize of \$100k was [awarded in 1997 with some fanfare](#) to the IBM Deep Blue team.

Ed also put up a prize for “math AI”, or, more specifically, automated theorem proving. It was [administered through the American Math Society](#) and a few “milestone prizes” were given out. But the grand Leibniz Prize “for the proof of a ‘substantial’ theorem in which the computer played a major role” was never claimed, the assets of the Fredkin Foundation withered, and the prize was withdrawn. (I wonder if some of the things done in the 1980s and 1990s by users of Mathematica should have qualified—but Ed and I never made this connection, and it’s too late now.)

Ed the Consultant

Particularly during his time at MIT, Ed did a fair amount of strategy consulting for tech companies—and Ed would tell me many stories about this, particularly related to IBM and DEC (which were in the 1980s the world’s two largest computer companies).

One story (whose accuracy I’ve never been able to determine) related to DEC’s ultimately disastrous decision not to enter the personal computer business. As Ed tells it, a team at DEC did a focus group about PCs—with Ken Olsen (CEO of DEC) watching. There was a young teacher in the group who was particularly enthusiastic. And Olsen seemed to be getting convinced that, yes, PCs were a good idea. As the focus group was concluding, the teacher listed off all sorts of ways PCs could change the world. But then, fatefully, he added right at the end: “And I don’t just mean here on Earth”. Ed claims this was the moment when Olsen decided to kill the PC project at DEC.

Ed told a story from the early 1970s about a giant IBM project called FS (for “Future Systems”):

IBM has this project. They’re going to completely revolutionize everything. The project is to design everything from the smallest computer to the new largest. They’re all to be multiprocessors. The specs were just fantastic. They promised to guarantee their customers 100% uptime. Their plans were, for instance, when you have a new OS, it’s updated. They guarantee 24-hour operation at all times. They plan to be able to update the OS without stopping this process. Things like that, a lot of goals that are very lofty, and so on.

Someone at IBM whom I knew very well, a very senior guy, came to me one day and said, “Look, these guys are in trouble, and maybe MIT could help them.” I organized something. Just under 30 professors of computer science came down to IBM. We got there on Sunday night and starting Monday morning, we got one lecture an hour, eight on Monday, Tuesday, Wednesday, Thursday, and four on Friday, describing the system. It was just spectacular, everything they were trying to do, but it was full of all kinds of idiocy. They were designing things that they’d never used. This whole thing was to be oriented about people looking at displays.

No one at IBM had done anything like that. They think, “Okay, you should have a computer display,” and they came up with certain problems that hadn’t occurred to the rest of us. If you’re looking at the display, how can you tell the difference between what you had put into the computer and what the computer had put in? This worried them. They came up with a hardware fix. When you typed, it always went on the right half of the screen; when the computer did something, it always went on the left half, or I may have it backwards, but that was the hardware.

...

What happened is I came to realize that they were so over their head in their goal that they were going to annihilate themselves with this thing. It was just going to be the world’s greatest fiasco for it. I started cornering people and saying, “Look, do you realize that you’re never going to make this work?” and so on, so forth. This came to the attention of people at IBM, and it annoyed them. I got a call from someone saying, “Look, you’re driving us nuts. We want to hear you out, so we’re going to conduct a debate.” There’s a guy named Bob [Evans], who was the head of the project. What happened was we’re in the boardroom with IBM, lots of officials there, and he and I have a debate.

I’m debating that they have to kill the project and do something else. He’s debating that they shouldn’t kill the project. I made all my points. He made all his points. Then a guy named [Mannie Piore](#), who was the one who thought of the idea of having a research laboratory, a very senior guy said to me, he said, “Hey, Ed,” he said, “We’ve heard you out.” He says, “This is our company. We can do this product even if you think we shouldn’t.” I said, “Yes, I admit that’s true.” He said, “You presented your case. We’ve heard you out, and we want to do it.” I said, “Okay.” He said, “Can you do us a favor?” I said, “What’s that?” He said, “Can you stop going around talking to people about why it has to be killed?” I said, “Look, I’ve said my piece. I’ve been heard out.” “Yes. Okay.” “I quit.”

I had only one ally in that room; that was John Cocke. As we were walking out of the room, he came over to me and said, “Don’t worry, Ed.” He said, “It’s going to fall over of its own weight.” I’ll never forget that. Ten days later, it was canceled. A lot of people were very mad at me.

I’m not sure what Ed was like as an operational manager of businesses. But he certainly had no shortage of opinions about how businesses should be run, or at least what their strategies should be. He was always keen on “do-the-big-thing” ideas. I remember him telling me multiple times about a company that did airplane navigation. It had put a certain

number of radio navigation beacons into its software. Ed told me he'd asked about others, and the company had said "Well, we only put in the beacons lots of people care about". Ed said "Just put all of them in". They didn't. And eventually they were overtaken by a company that did.

Ed the Businessman

Ed's great business success—and windfall—was III. But Ed was also involved with a couple dozen other companies—almost all of which failed. There's a certain charm in the diversity of Ed's companies. There was Three Rivers Computer Corporation, that made the PERQ computer. There was Triadex, that made the Muse. There was a Boston television station. There was an air taxi service. There was Fredkin Enterprises, importing PCs into the Soviet Union. There was Drake's Anchorage, the resort on his island. There was Gensym, a maker of AI-oriented process control systems, which was a rare success. And then there was Reliable Water.

Ed's island—like many tropical islands—had trouble getting fresh water. So Ed decided to invent a solution, coming up with a new, more energy-optimized way to do reverse osmosis—with a dash of AI control. Reliable Water announced its product in May 1987, desalinating water taken from Boston Harbor and serving it to journalists to drink. (Ed told me he was a little surprised how willingly they did so.)

D10

L

THE NEW YORK TIMES, WEDNESDAY, MAY 13, 1987

BUSINESS TECHNOLOGY

Water From Sea: A Cheaper Way

By CALVIN SIMS

RESearchers here have developed a new process that can extract fresh drinking water from sea water at a significantly lower cost than can existing desalination systems.

The process, announced yesterday at the small waterfront development laboratory of the Reliable Water Company, employs three different technologies: a filtration technique known as reverse osmosis, an energy recovery system and a computer that runs an artificial intelligence program.

Analysts say the system is highly automated, thus reducing maintenance, and uses far less energy than conventional desalination technologies. Officials of Reliable Water said the new system could produce water suitable for drinking or industrial uses for one-third to two-thirds the cost of conventional desalination systems.

"I've been in this business 15 years and I can say without doubt that this design represents the next generation of desalination systems," said Kent N. Nielsen, president of the Nielsen Consulting Company, which evaluates water treatment processes for

resorts and other clients in the Caribbean.

"And it tastes pretty good, too," he added, sipping a glass of water that had just been pumped from brackish Boston Harbor and made drinkable by a prototype of the new system.

Sea water desalination is already a \$1 billion market worldwide and is expected to grow steadily in the next decade, analysts said. Middle Eastern countries, particularly Kuwait, Saudi Arabia, the United Arab Emirates, Qatar and Bahrain account for 65 percent of the market, and the United States for 10 percent.

The market's growth in the next decade, the analysts said, will come from the replacement of worn-out and obsolete equipment in existing plants, and from the installation of new plants in arid regions where the demand is rising.

Dr. Jim Birkette, a senior analyst for Arthur D. Little and an expert in desalination technology, said there was major growth potential in the Gulf states, particularly Florida, where the population was increasing and putting stress on aquifers and wells.

In addition, he said, the demand for desalination systems on islands, where there are few fresh water resources, has remained steady.

"A medium-sized, highly efficient

Using hydraulic oil pumps, transfer barriers and special valves, the Reliable Water system transfers the fluid pressure of the waste brine to incoming sea water, substantially reducing the energy requirements.

A computer using artificial intelligence techniques controls all aspects of the system, including valve adjustments, sensor readings and maintenance management.

system like Reliable Water's should have great promise in these areas," Dr. Birkette said.

Most current desalination plants employ a technology called multi-stage flash distillation, which removes contaminants from sea water by boiling it and condensing the steam.

The new system was invented by Edward Fredkin, a former professor of computer science at the Massachusetts Institute of Technology. Mr. Fredkin said he decided to enter the field after the desalination machine he had purchased for an island resort continually malfunctioned.

"It became clear to me then that this industry needed a more rigorous system that produces fresh water at less cost," he said.

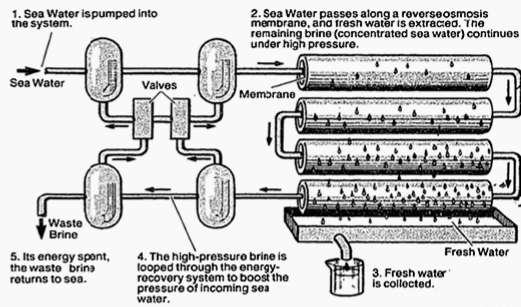
The first part of the Reliable Water process is a filtration technique known as reverse osmosis. In this method, highly pressurized sea water is pumped into the system and passed through a semipermeable membrane that allows only fresh water to flow through.

Once the fresh water has been extracted, salt and other contaminants are left behind in the brine, or concentrated sea water.

As a distinct unit, a reverse osmosis system is expensive to operate because of the substantial energy needed to constantly pressurize incoming sea water.

However, the Reliable Water system uses an energy recovery mechanism that recycles most of the fluid

The Desalination Process Using Energy Recovery and Artificial Intelligence Control



The New York Times/Photos by Joe Wilson, drawing by Hank Ren

pressure from the brine as it leaves the system. Using hydraulic oil pumps, transfer barriers and special valves, the system extracts the pressure in the brine and transfers it to the incoming sea water, substantially reducing the energy requirements.

The new desalination system is controlled by a sophisticated artificial intelligence program that runs on a standard I.B.M. personal computer.

The program, which was designed by the Gensym Corporation, a Boston-based software company, monitors dozens of sensors that detect valve pressures, eliminating the cost of human attendants. The program compares the actual performance of the Reliable Water system with that of a theoretically ideal system, and if significant discrepancies arise the computer sends a telephone signal to maintenance workers.

"This is the first sophisticated reverse osmosis system to employ both artificial intelligence and energy saving hydraulics," said Francis E. Stone, president of Stone & Associates, an engineering consulting firm with interests in desalination and membrane technology.

The new system has attracted a list of prominent investors, among them Dr. Jerome Wiesner, former president of M.I.T., and Gordon Bell, former engineering vice president of the Digital Equipment Corporation, who is now assistant director of the National Science Foundation.

"The appeal that this system had for me was the cost benefit factor of almost two," Mr. Bell said. "If you have that and patent protection, then you have a breakthrough."

Reliable Water plans to market its desalination process in commercial

plants that would produce about 30,000 gallons of water a day, or enough to supply a small hotel. The company is scheduled to have its first plant in operation in Boston by the first quarter of 1988, after which it plans to build similar operations in the Caribbean, the Middle East and Asia.

Mr. Fredkin said that initially the company would form partnerships in which Reliable Water would build desalination plants and customers would purchase water either for drinking or other industrial purposes.

There are more than 100 foreign and domestic companies with interests in desalination products and technology. Among the larger players are Fafnura and Mitsubishi, both of Japan, Sidem of France, Aquo Chem of Milwaukee, and Hydranautics, a division of Roman Hosi.



Edward Fredkin, left, toasts William Choate of the Reliable Water Company with a glass of desalinated water from Boston Harbor.

Advances in Ovens Chefs Taking Toques Off to Triple Unit



Looking at my archives I see I was sufficiently charmed by the picture of Ed posing with his elaborate "intelligent" glass tubing that I kept the article from *New Scientist*:

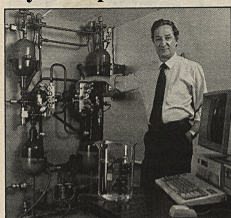
Expert systems put sea water on tap

A COMPANY in Boston is using artificial intelligence techniques to save energy when desalinating water. The company made its biggest splash at a press conference last week by handing out bottles of purified Boston Harbor water. "There was none of the salt, selenium, DDT, bacteria or any of the other poisons that make Boston Harbor the nation's dirtiest," reported the *Boston Globe*. "Served well chilled it [the purified water] tasted just fine."

The Reliable Water Company uses a modified version of the reverse osmosis process that is common in other desalination plants. The modifications cut energy usage—the major contributor to costs—by one-third to two-thirds, says Bill Choate, vice president of engineering. The company believes it can produce drinkable water for less than one cent a gallon (1 US gallon = 3.79 litres), much more expensive than ordinary city water, but much cheaper than other desalinated water.

In reverse osmosis, a semi-permeable membrane removes pure water from brine or sea water. Pure water can pass through the membrane, but not salt or pollutants. If both sides of the membrane are at the same pressure, pure water on one side will diffuse into salt water on the other side. However, applying high pressures—900 to 1000 pounds per square inch—to the salt water forces pure water to trickle through to the fresh-water side, where pressure is near or slightly above atmospheric pressure (14 pounds per square inch).

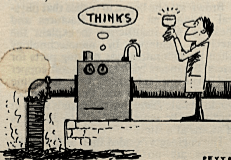
Applying the pressure needed to force fresh water through the membrane consumes most energy used by reverse osmosis. Typically yields are 2 to 4 litres of fresh water for every 10 litres of input water; the rest emerges as waste brine with elevated salt concentration. The waste brine also is at high pressure, but many desalination plants just drain it off and lose that energy. Some desalination plants run



Fredkin's "intelligent" desalination system

high-pressure brine through a turbine to recover part of the energy, but that process is inefficient. The Reliable Water process uses a hydraulic system to recycle pressure from the waste brine back to the input water.

The hydraulic system incorporates a series of pressure barriers and valves to redirect the pressure. Choate compares it to rubber balloons which force sea water into the processor at high pressures. Proper operation of the system requires active control, which is where artificial intel-



ligence techniques enter the picture.

Reliable Water's prototype system includes an IBM AT personal computer running an "expert system" program with a large knowledge base, which is linked to many sensors and controls. "It essentially watches the system and anticipates problems," explains Choate. He adds, "As far as we know, it will be the first real time expert system in [commercial] operation." It uses rules and information stored in the expert system to deduce when cleaning or adjustments are needed. If a sensor fails, the program can decide if the sensor is critical to system function, or if it can continue operating. It will be able to develop its own rules as well as using pre-programmed routines.

One goal is to simplify operation, easing requirements for operator training. The software will include "the whole operating manual," Choate says. "When it is time to repair [the system], the computer will automatically cycle all the valves into the proper position, so the operator doesn't have to do a lot of thinking." That could be very important if the system is used in remote areas where trained technicians are not available.

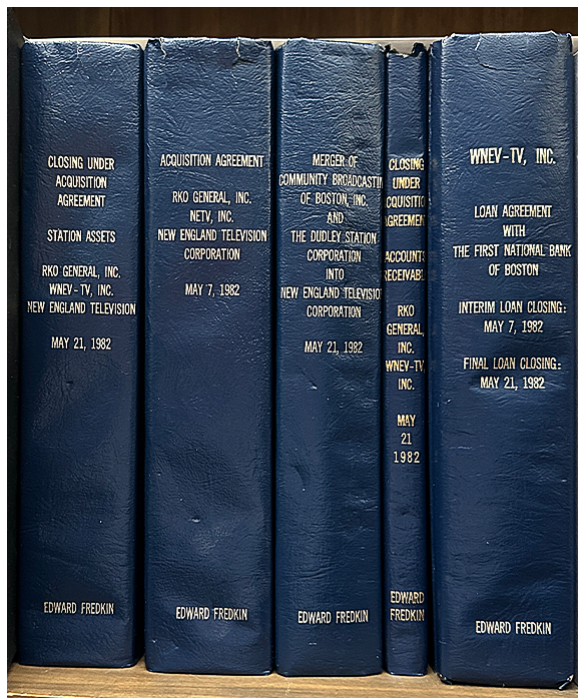
An in-house prototype produces 4500 litres of water a day, and the company is working on a commercial model that will produce 110,000 to 150,000 litres a day. Plans are to complete the first commercial version this year, and install it by early next year. One potential market is the Caribbean, where high energy costs make energy savings very attractive. Another is the Middle East, where energy costs are modest, but where the expert system could alleviate problems of technical support.

Reliable Water has an impressive list of backers. Its founder and chairman, Edward Fredkin, is a former head of the computer-science laboratory at the Massachusetts Institute of Technology.

As Ed told it to me, Reliable Water was just about to sell a major system to an Arab country

when his well-pedigreed CEO somehow cheated him, and the deal fell through.

But what about the television station? How did Ed get involved with that? Apparently in 1969 Jerry Wiesner, then president of MIT, encouraged Ed to support a group of Black investors (led by a certain Bertram Lee) who were challenging the broadcasting license of Boston’s channel 7. Years went by, other suitors showed up, and litigation about the license went all the way to the Supreme Court (which described the previous licensee as having shown an “egregious lack of candor” with the FCC). For a while it seemed like channel 7 might just “go dark”. But in early January 1982 (just a couple of weeks before I first met him) Ed took over as president of New England Television Corporation (NETV)—and in May 1982 NETV took over channel 7, leaving Ed with a foot of acquisition documents in his home library, and a television channel to run:



There’d been hopes of injecting new ideas, and adding innovative educational and other content. But things didn’t go well and it wasn’t long before Ed stepped down from his role.

A major influence on Ed’s business activities came out of something that happened in his personal life. In 1977 Ed had been married for 20 years and had three almost-grown children. But **then he met Joyce**. On a flight back from the Caribbean he sat next to a certain Joyce Wheatley who came from a prominent family in the British Virgin Islands and had just graduated with a BS in economics and finance from Bentley College (now Bentley University) in Waltham, MA. As both Ed and Joyce tell it, Ed immediately gave advice like

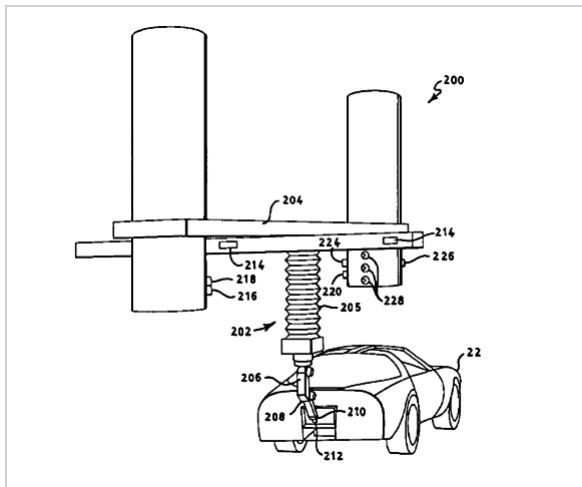
that the best way to overcome a fear of flying was to learn to fly (which much later, Joyce in fact did).

Joyce was starting work at a bank in Boston, but matters with Ed intervened, and in 1980 the two of them were married in the Virgin Islands, with Feynman serving as Ed's best man (and at the last minute lending Ed a tie for the occasion). In 1981, Ed and Joyce had a son, who they named Richard after Richard Feynman (though now themed as "Rick")—of whom Ed was very proud.

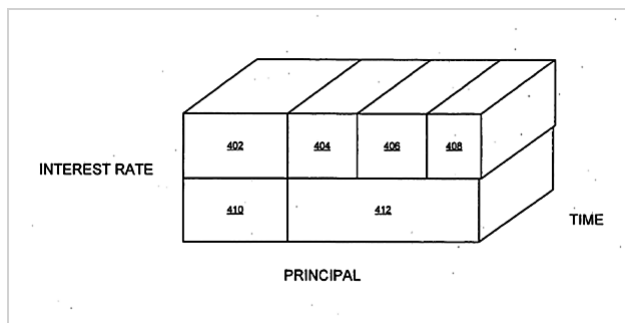
When Ed died, Joyce and he had been married for 43 years—and Joyce had been Ed's key business partner all that time. They made many investments together. Sometimes it'd start with a friend or vendor. Sometimes Ed (or Joyce) would meet students or others—who'd be invited over to the house some evening, and leave with a check. Sometimes the investments would be fairly hands-off. Sometimes Ed would get deeply involved, even at times playing CEO (as he did with Three Rivers and NETV).

When the web started to take off, Ed and Joyce created a company called Capital Technologies which did angel investing—and ended up investing in many companies with names like Sourcecraft, SqueePlay, EchoMail, Individual Inc. and Radnet. And—like so many startups of this kind—most failed.

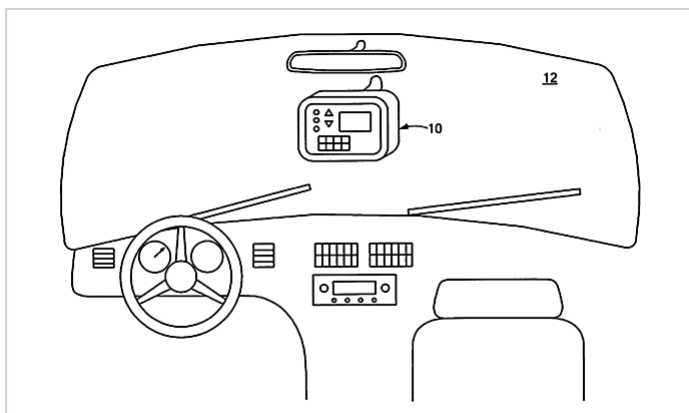
Ed also continued to have all sorts of ideas of his own, some of which turned into patents. And—like so much to do with Ed—they were eclectic. In 1995 (with a couple of other people) there was one based on using evanescent waves (essentially photon tunneling) to more accurately find the distance between the read/write head and the disk in a disk drive or CD-ROM drive. Then in 1999 there was the "[Automatic Refueling Station](#)"—using machine vision plus a car database to automate pumping gas into cars:



That was followed in 2003 by a patent about securely controlling telephone switching from web clients. In 2006, there was a patent application named simply “[Contract System](#)” about an “algorithmic contract system” in which the requirements of buyers and sellers of basically anything would be matched up in a kind of tiling-oriented geometrical way:



In 2011 there was “[Traffic Negotiation System](#)”, in which cars would have rather-airplane-like displays installed that would get them in effect to “drive in formation” to avoid traffic jams:



Ed’s last patent was filed in 2015, and was essentially for a scheme to cache large chunks of the web locally on a user’s computer—a kind of local CDN.

But all these patents represented only a small part of Ed’s “idea output”. And for example Ed told me many other tech ideas he had—a few of which I’ll mention later.

And Ed’s business activities weren’t limited to tech. He did his share of real-estate transactions too. And then there was his island. For years Joyce and Ed continued to operate Drake’s Anchorage, and tried to improve the infrastructure of the island—with Ed, as Joyce tells it, more often to be found helping to fix the generator on the island than partaking of its beaches.

Back in 1978 Ed had acquired a “neighbor” when Richard Branson bought Necker Island, which was a couple of miles further out towards the Atlantic than Moskito Island. Ed told me quite a few stories about Branson, and for years had told me that Branson wanted to buy his island. Ed hadn’t been interested in selling, but eventually agreed to give Branson right of first refusal. Then in 2007 a Czech (or were they a Russian?) showed up and offered to buy the island for cash “to be delivered in a suitcase”. It was all rather sketchy, but Ed and Joyce decided it was finally time to sell, and let Branson exercise his right of first refusal, and buy the island for about \$10M.

Ed and His Toys

Ed liked to buy things. Computers. Cars. Planes. Boats. Oh, and extra houses too (Vermont, Martha’s Vineyard, Portola Valley, ...)—as well as his island. Ed would typically make decisions quickly. A house he drove by. New tech when it first came out. He was always proud of being an early adopter, and he’d often talk almost conspiratorially about the “secret” features he’d figured out in new tech he’d bought.

But I think Ed’s all-time favorite “toys” were planes—and over the course of his life he owned a long sequence of them. Ed was a serious (and, by all reports, exceptionally good) pilot—with an airplane transport pilot license (plus seaplane and glider licenses). And I always suspected that his cut-and-dried approach to many things reflected his experience in making decisions as a pilot.

Ed at different times had a variety of kinds of planes, usually registered with the vanity tail number N1EF. There were twin-propellor planes. There were high-performance single-

propellor planes. There was the seaplane that I’d “met” in the Caribbean. At one time there was a jet—and in typical fashion Ed got himself certified to fly the jet singlehandedly, without a copilot. Ed had all sorts of stories about flying. About running into Tom Watson (CEO of IBM) who was also a pilot. About getting a new type of plane where he thought he was getting #5 off the production line, but it was actually #1—and one day its engine basically melted down, but Ed was still able to land it.

Ed also had gliders, and competed in gliding competitions. Several times he told me a story—as a kind of allegory—about another pilot in a gliding competition. Gliders are usually transported with their wings removed, with the wings attached in order to fly. Apparently there was an extra locking pin used, which the other pilot decided to remove to save weight, because it didn’t seem necessary. But when the glider was flying in the competition its wings fell off. (The pilot had a parachute, but landed embarrassed.) The very pilot-oriented moral as far as Ed was concerned: just because you don’t understand why something is there, don’t assume it’s not necessary.

Ed and the Soviet Union

One of the topics about which Ed often told “you-can’t-make-this-stuff-up” stories was the Soviet Union. Ed’s friend John McCarthy had parents who were active communists, had learned Russian, and regularly took trips to the Soviet Union. And as Ed tells it McCarthy came to Ed one day and said (perhaps as a result of having gotten involved with a Russian woman) “I’m moving to the Soviet Union”, and talked about how he was planning to dramatically renounce his US citizenship. McCarthy began to make arrangements. Ed tried to talk him out of it. And then it was 1968 and the Soviets send their tanks into Czechoslovakia—and McCarthy is incensed, and according to Ed, sends a telegram to a very senior person in the Soviet Union saying “If you invade Czechoslovakia then I’m not coming”. Needless to say, the Soviets ignored him. Ed told me he’d said at the time: “If the Russians were really smart and really understood things, and they had to choose between John McCarthy and Czechoslovakia, they should have chosen John McCarthy.” (McCarthy would later “flip” and become a staunch conservative.)

Perhaps through McCarthy, Ed started visiting the Soviet Union. He didn’t like the tourist arrangements (required to be through the government’s Intourist organization)—and

decided to try to do something about it, sending a survey to Americans who'd visited the Soviet Union:

EDWARD AND DOROTHY FREDKIN

166 Hyslop Road
Brookline, Massachusetts 02146

April 26, 1976

Dear Survey Participant:

I am trying to gather information from a meaningful sample of United States visitors to the Soviet Union. I have placed ads in leading papers across the country, and have received about 500 responses. It is my hope that most of you will be willing to take a few minutes to fill out the enclosed questionnaire. Each participant will, if he so requests, receive a summary of the results.

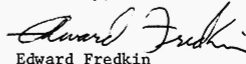
My purpose in conducting this survey is in the hope of bettering relations between the United States and the Soviet Union by bringing certain facts to the attention of Intourist management, and other parts of the Soviet Union's government, that may result in beneficial changes in policy and procedures. I simply want to let Intourist know how it can better its services. This came about because of a recent trip to the Soviet Union which seemed to reveal serious flaws in some of Intourist's procedures.

In particular, it appears that it is often difficult to obtain a refund for services pre-paid in the United States, but not delivered in the Soviet Union. In some cases, services seem to be sold when it should be known at the time of sale that these services cannot possibly be delivered (such as car rentals and tours).

I have held discussions with Intourist officials in the Soviet Union, and feel that the next step, if anything is to be done about bettering the situation, is to get opinions from others who have travelled to the Soviet Union, since, admittedly, my personal sample is not statistically meaningful.

I want to sincerely apologize for the great delay in getting back to you, and hope that you are still interested in helping me in this endeavor.

Sincerely,

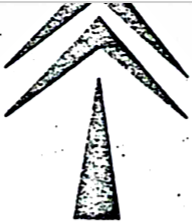


Edward Fredkin

A year later, Ed was back in the Soviet Union, attending a somewhat all-star conference (along with McCarthy) on AI—with a rather modern-sounding collection of topics:

МЕЖДУНАРОДНОЕ СОВЕЩАНИЕ
ПО ИСКУССТВЕННОМУ ИНТЕЛЛЕКТУ

18—24 апреля 1977
Репино, Ленинград, СССР



INTERNATIONAL MEETING
ON ARTIFICIAL INTELLIGENCE

18-24 April, 1977
Repino, Leningrad, USSR

The following topics will be covered:

Theories of knowledge-representation
 Acquisition and measurement of knowledge
 Machine aids to mathematical reasoning
 Computer game-playing
 Intelligent programs as scientists' assistants
 Visual perception in man and machine
 Manipulation and locomotion in robot devices
 Automatic plan-formation and program synthesis
 Question-answering systems and database design
 Programs for understanding natural language
 Impact of Machine Intelligence on computer science

There will in addition be laboratory visits and demonstrations.

Other non-Soviet contributors include (titles of their contributions are given where known):

Sh. Aida, Electrocommunications Institute, Tokyo, Japan.

J.M. Foster, Royal Radar Establishment, Malvern.

E. Fredkin, Massachusetts Institute of Technology, U.S.A.

J. McCarthy, Stanford University, U.S.A.: "Epistemological problems of A.I."

D. Michie, University of Edinburgh: "Vision and chess research at Edinburgh".

J. Moore, Stanford Research Institute, U.S.A.

N. Nilsson, Stanford Research Institute, U.S.A. "Current trends in problem-solving".

R.J. Popplestone, University of Edinburgh.

E. Sandewall, Linköping University, Sweden.

Here's a photograph of a bearded Ed in action there—with a very Soviet simultaneous translation booth behind him:



Ed used to tell a story about Soviet computers that probably came from that visit. The Soviet Union had made a copy of an IBM mainframe computer—labeling it as a “RYAD” computer. There was a big demo—and the computer didn’t work. The generals in charge asked “Well, did you copy everything?” As it turned out, there was active circuitry in the “IBM” logo—and that needed to be copied too. Or at least that’s what Ed told me.

But Ed’s most significant interaction with the Soviet Union came in the early 1980s. The US had in place its [CoCom list](#) that embargoed export of things like personal computers to the Soviet Union. Meanwhile, within the Soviet Union, photocopiers were strictly controlled—to prevent non-state-sanctioned flow of information. But as Ed tells it, he hatched a plan and sold it to the Reagan administration, telling them: “You’re on the wrong track. If we can get personal computers into the Soviet Union, it breaks their lock on the flow of information.” But the problem was he had to convince the Soviets they wanted personal computers.

In 1984 Ed was in Moscow—supposedly tagging along to a physics conference with an MIT physicist named [Roman Jackiw](#). He “dropped in” at the Computation Center of the Academy of Sciences (which, secretly, was a supplier to the KGB of things like speech recognition tech). And there he was told to talk to a certain [Evgeny Velikhov](#), a nuclear physicist who’d just been elected vice president of the Academy of Sciences. Velikhov arranged for Ed to give a talk at the Kremlin to pitch the importance of computers, which

apparently he successfully did, after convincing the audience that his motivation was to make the world a safer place by balancing the technical capabilities of East and West.

And as if to back up this point, while he was in the Soviet Union, Ed wrote a 5-page piece from “A Concerned Citizen, Planet Earth” addressed “To whom it may concern” in Moscow and Washington—ending with the suggestion that its plan might be discussed at an upcoming meeting between Andrei Gromyko and Ronald Reagan at the UN:

A Concerned Citizen
Planet Earth

13 September, 1984

To whom it may concern
USSR, Moscow

To whom it may concern
Washington, USA

The world is rapidly plunging towards a terrible fate, which may involve the destruction of a majority of mankind, and the fruits of thousands of years of civilization. Why? That is a question that deserves an answer.

Every creature that lives on this planet has a will to survive. Yet, no creature survives beyond its normal life span because all creatures are mortal. While we may want to live forever, it is certain that we will all die someday. Under the best and most peaceful of circumstances, we can expect that every one now alive on this planet will be dead by the year 2150.

Every species of creature on this planet has survived for thousands of years and also has the ability to survive for thousands of years into the future. Yet no species has the will to survive, because a species cannot have a will of its own. If mankind manages to continue surviving on this planet, it will not be a matter of the will of our species, but because of the will of the individuals who wish to have mankind survive. I and others like me, who want to survive, to have others survive, to have our species survive must express our will against all forces that threaten that survival. We need not care if one man so wants to win that he puts winning ahead of his own survival. We all must care when some want to win, to have their nation win, and they put that goal ahead of the survival of all mankind.

The fear of annihilation knows no national boundaries. The will to survive knows no national boundaries. Those who wish to go on living, whoever and wherever they may be, must express their will in terms of actions that are effective. Throughout history, governments have proven themselves unable to avoid war. Such wars are usually preceded by an increase in tensions and an increase in hostility. In fact, situations similar to those developing today, between the USSR and the USA are very much like those that have led to war in the past. The difference is that today, everyone in the world is threatened by this USA-USSR confrontation. It is those who are threatened who must work to save themselves and their descendants by finding ways to reduce the chance of war. Since everyone is threatened, we all must work on this problem. There is no aspect that is too small to be worthy of careful and considerate effort. It is not a time for us to be patient, rather it is a time for us to be purposeful and persistent.

The piece mentions another issue: the fate of prominent, but by then dissident, Soviet physicist [Andrei Sakharov](#), who was in internal exile and reportedly on hunger strike. Ed hatched a kind of PCs-for-Sakharov plan in which the Soviets would get PCs if they freed Sakharov.

Meanwhile, in true arms-dealer-like fashion, he'd established Fredkin Enterprises, S.A. which planned to export PCs to the Soviet Union. He had his student Norm Margolus spend a summer analyzing the CoCom regulations to see what characteristics PCs needed to have to avoid embargo.

In the Reagan Presidential Library there's now a fairly extensive file entitled "[Fredkin Computer Exports to USSR](#)"—which for example contains a memo reporting a call made on August 25, 1984, by then-vice-president George H. W. Bush to Sakharov's stepdaughter, who was by that time living in Massachusetts (and, yes, Ed was described as a "PhD in computer science" with a "flourishing computer business"):

August 25, 1984
Kennebunkport, ME

MEMORANDUM FOR RECORD

FROM: Bill Eckert *WDE*

SUBJECT: Telephone Call, 0920 EDT, Sat 25 Aug 84
Vice President George Bush to Mrs. Tanya Yankeleyevich (sp?),
daughter of Elaine Bommer. Her phone # in Mass. is (617) 964-8528.

AUG 25 A10: 57

At the request of Gordon Zacks (by phone Aug 25), the Vice President called Tanya to receive a request from her. After noting that there is a letter coming to the Vice President from the family, she asked for advice on the following situation.

A Dr. Fradkin will depart the US today or tomorrow for Moscow at the invitation of the Soviet Academy of Sciences. He is a Ph.D. in computer science, a professor at MIT, and owner of a flourishing computer business. He would like to know if the Administration would like for him to ask if the Soviets are interested in obtaining anything from the US in exchange for Sakharov.

Tanya said, "I do not think it would be wise to give the Soviets a definite impression of what would be available to them, but just to tell Dr. Fradkin that it would be all right for him to ask the Soviets what they might want in return."

Tanya said Fradkin lives in Massachusetts near her. His home phone is (617) 277-4444. "If he has left for Moscow already, surely the US Embassy there could contact him if necessary."

Soon the White House is communicating with the US embassy in Moscow to get a message to Ed:

DECLASSIFIED
 NLRR 606-714/7 #9835
 BY CU NARA DATE 7/7/08

~~CONFIDENTIAL~~

6677

File
OUTGOING

WHITE HOUSE SITUATION ROOM

PAGE 01 OF 02 THE WHITE HOUSE 6966 DTG: 082109Z SEP 84 PSN: 075845
 SIT046 AN004465 TOR: 252/2036Z

DISTRIBUTION: RLPH MATL SHAN /003

PRIORITY
 DE RUEADWW #6966 2522036
 P 082109Z SEP 84 ZFF-1 ZFF-4
 FM THE WHITE HOUSE

TO AMEMBASSY MOSCOW

INFO SECSTATE

~~CONFIDENTIAL~~ WH06966

TO CHARGE KAMMAN FROM AMBASSADOR JACK MATLOCK

PLEASE CONVEY FOLLOWING MESSAGE TO EDWARD FREDKIN FROM ME:
 QUOTE
 DEAR EDWARD:

THANK YOU FOR THE MESSAGE REGARDING YOUR TALKS IN MOSCOW.

YOUR REPORT ON YOUR CONVERSATIONS WITH THE ACADEMY OF SCIENCES WAS MOST INTERESTING, AND I THINK YOUR COMMENTS ON THE FUTURE IMPLICATIONS OF THE INTRODUCTION OF LARGE NUMBERS OF MICROCOMPUTERS INTO SOVIET SOCIETY ARE QUITE PERSUASIVE. THE QUESTION OF EXPORT LICENSING OF MANUFACTURING FACILITIES, SHOULD IT ARISE IN THE FUTURE, WOULD OF COURSE REQUIRE DETAILED EXAMINATION IN THE U. S. GOVERNMENT. HOWEVER, ULTIMATE APPROVAL IS CERTAINLY NOT BEYOND THE REALM OF THE POSSIBLE. OBVIOUSLY, ANY DECISION ON A MAJOR QUESTION OF THIS SORT WOULD BE INFLUENCED BY THE OVERALL POLITICAL RELATIONSHIP AT THE TIME. WHILE I DO NOT WISH TO IMPLY ANY SORT OF DIRECT LINKAGE, IT IS CLEAR THAT THE SAKHAROV SITUATION DOES HAVE A BEARING ON THE POLITICAL CLIMATE, AND A HUMANITARIAN RESOLUTION OF THAT TRAGIC SITUATION WOULD CONTRIBUTE TO ITS IMPROVEMENT. I WOULD HOPE THAT THIS POINT IS UNDERSTOOD BY YOUR SOVIET INTERLOCUTORS.

AS FOR YOUR SPECIFIC REQUEST, WE WOULD LIKE TO BE HELPFUL IF WE COULD, GIVEN OUR SHARED INTEREST IN RESOLVING THE SAKHAROV TRAGEDY. HOWEVER, IT IS NOT OUR POLICY TO MAKE EXPLICIT OR IMPLICIT

MATLOCK

And things are quickly starting to sound as if they were from a Cold War spy drama (there's no evidence Ed was ever officially involved with the US intelligence services, though):



United States Department of State

Washington, D.C. 20520

~~CONFIDENTIAL~~

September 11, 1984

TO: NSC - Jack Matlock
FROM: EUR/SOV - Tom ^{WA}Simons
SUBJECT: Reply from Edward Fredkin

Curt Kamman passed to Edward Fredkin your response to Fredkin's September 1 message. Fredkin still wishes to pursue the Sakharov problem and he has asked that we transmit to you the attached reply outlining his proposals on this issue. We will also need to respond to Fredkin's inquiry concerning contact with you from Moscow via the secure telephone.

Attachment:
As stated.

I don't think Ed ever ended up talking to Sakharov, but on November 6, 1984, Fredkin Enterprises was sent a letter by Velikhov ordering 100 PCs for the Academy of Sciences, and saying they hoped to order 10,000 more. But the US was not as speedy, and in 1985 there was still back and forth about CoCom issues. Ed of course had a plan:

Moscow
May 22, 1985

The following proposal was conceived by Academician E. P. Velikhov, Vice President of the Academy of Science of the USSR, and further developed by Academician Velikhov and Professor Edward Fredkin of MIT.

Today there are millions of personal computers in the world, and the number is growing rapidly. Within a decade, personal computers will be able to revolutionize all aspects of intellectual work. The situation is similar to the industrial revolution when science and engineering allowed for the development of machinery and engines that revolutionized all aspects of manual labor. The industrial revolution lasted for more than a century.

As a concrete first step, we propose that a meeting be held at the earliest convenient time. This meeting should last for 7 to 10 days, and should be attended by no more than 12 participants and two observer-moderators. From the United States, the 6 participants should be selected to represent industry, education, and government. From the Soviet side, the 6 participants should also be selected using similar criteria.

Through the experience of learning a subject while using software from another country, schoolchildren will gain understanding and appreciation of the peoples of foreign countries. Those from different countries, who work together to create better software, will learn from each other, and along the way, bonds of friendship and cooperative spirit will be forged. It is hard to imagine a more constructive or less controversial project that can capture the imagination and enthusiasm of peoples from both the United States and the Soviet Union.

Sincerely,

Academician E. P. Velikhov
Academy of Sciences
Moscow
USSR

Professor Edward Fredkin
Massachusetts Institute of Technology
Cambridge, Massachusetts
USA

And indeed in the end Ed did succeed in shipping at least some computers to the Soviet Union, adding a hack to support Cyrillic characters. Ed often took his family with him to Moscow, and he told me that his son Rick created quite a stir when at age 6 he was seen there playing a game on a computer. Up to then, computers had always been viewed as expensive tools for adults. But after Rick's example there were suddenly all sorts of academicians' kids using computers.

(In the small world that it is, one person Ed got to know in the Academy of Sciences was a certain [Arkady Borkovsky](#)—who in 1989 would leave Russia to come work at our company, and who would later co-found Yandex.)

By the way, to fill in a little color of the time, I might relate a story of my own. In 1987 I went to a (rather Soviet) conference in Moscow on “Logic, Methodology and Philosophy of Science.” Like everyone, I was assigned a “guide”. Mine continually tried to pump me for information about the American computer industry. Eventually I just said: “So what do you actually want to know?” He said: “We’ve cloned the Intel 8086 microprocessor, and we want to know if it’s worth cloning the Motorola 68000. Motorola has put a layer of epoxy that makes it hard to reverse engineer.” He assumed that the epoxy was at the request of the US government, to defeat Soviet efforts—and he didn’t believe me when I said I thought it was much more likely there to defeat Intel.

Ed told me another story about his interactions with Soviet computer efforts after Gorbachev came to power:

Before the days of integrated circuits the way IBM and Digital built computers was they put the whole computer together, and then it would sit for six weeks in “system integration” while they made the pieces work together and slowly got the bugs out.

The Russians built computers differently because that seemed logical to them. They’d send all the components down there and then some guy was supposed to plug them together, and they were supposed to work. But they didn’t. With these big computers, they never made any of them work.

The Academy of Sciences had one. And one time I went to see their big computer, so they unlock the doors to this dusty room where the computer is, where it’s not being used because it doesn’t work, and all this information is being kept secret, not from the United States, but from the leadership. When I discovered all this I documented it ... and I wrote a 40-page document that explained it.

I was making trips with Rick often and Mike [his older son] very often. On one trip when I arrived, they tell me, “Oh, you have to come to this meeting.”

I don’t speak Russian. I never knew it. I’m seated at this meeting, and there’s a Russian friend of mine [head of the Soviet Space Research Institute] next to me. We’re just sitting there, and things are going on. I still don’t know what that meeting was, but I had this 40-page document. I gave it to my friend. He starts reading. He says, “Oh, this is so interesting.” It got to be about ten o’clock at night and they said, “Everyone come back in the morning. Nine o’clock.”

My friend said, “Can I borrow this [document]? I’ll bring it back in the morning”. I said, “Sure, go ahead.” He comes back next morning. He says to me, “I have good news, and I

have bad news.” I said, “What’s the good news?” He says, “Your document has been translated into Russian.” I said, “You left here with a 40-page typewritten document. I don’t believe you.” He said, “Well, my institute recently took on the task of translating scientific American into Russian.

“When I left here, I went to my institute, called in the translators, and they all came in. We divided the document up between them, and it’s all been translated into Russian.”

The document was the analysis of the RYAD situation with the recommendation that the only thing they could do was to cancel it all.

I said, “Okay, what’s the bad news?” He says, “The bad news is it’s classified secret.” When you made a copy or did something, you had to have a government person look at it. They classified it. I said to him, “You can’t classify my documents.” He said, “Of course not. We haven’t. It’s just the Russian one that’s secret.”

Then maybe a week later, he said, “Gorbachev’s read your document.” He canceled it. RYAD. Some people I know were looking to kill me.

In Moscow, there’s a building that’s so unusual. It’s on a highway leading into the city. It’s about five stories high. It’s about a kilometer long, okay? It’s a giant building. I was in it a few years ago, and it’s just a beehive of startups, almost all software startups. That was the RYAD Software Center, okay? 100,000 people got put out of work.

Ed Becomes a Physics Professor

When I first met Ed in 1982, he was in principle a professor at MIT. But he was also CEOing a computer company (Three Rivers), and, though I didn’t know it at the time, had just become president of a television channel. Not to mention a host of other assorted business activities. MIT had a policy that professors could do other things “one day a week”. But Ed was doing other things a lot more than that. Ed used to say he was “tricked” out of his tenured professorship. Because in 1986 he was convinced that with all the other things he had going on, he should become an adjunct professor. But apparently he didn’t realize that tenure doesn’t apply to adjunct professors. And, as Ed told it, the people in the department considered him something of a kook, and without tenure forcing them to keep him, were keen to eject him.

Minsky’s neighbor in Brookline, MA, was a certain Larry Sulak—the very energetic chairman of the physics department at Boston University (and someone I have known since the 1970s). Ed knew Sulak and when Ed was ejected from MIT, Sulak seized the opportunity to bring Ed in as a physics professor at Boston University. Sulak asked me to

write a letter about Ed (and, yes, particularly after the research for this piece, there are some things I would change today):

Subject: Re: Ed Fredkin
Date: Aug 24, 1988
From: Stephen Wolfram
To: Larry Sulak

Dear Larry:

In this century, people like Ed Fredkin have been very rare. Ed Fredkin is a gentleman scientist. He has made several fortunes in business, yet he chooses to spend much of his time thinking about science.

The main thing he thinks about is what ideas from computing can tell us about physics. This is an area that I believe has fundamental importance for physics. There are many issues about the behaviour of complex physical systems where the best hope for analysis and understanding comes from computational ideas. There are also many traditional problems in quantum physics and other fundamental areas that I suspect are most likely to be solved by thinking about things from a computational point of view.

Ed Fredkin has had some very good ideas about physics and its relation to computation. Probably the single most important was his independent discovery of the possibility of thermodynamically reversible computation. von Neumann got this wrong — by thinking about things from a computational point of view, Fredkin got it right.

Fredkin has been convinced for many years that cellular automata — basically computational models — could describe fundamental physical processes. As you know, I have worked on using cellular automata to model various specific physical processes. Fredkin is trying to do something grander — he wants to show that all of physics can be reproduced by a cellular automaton. If he is right the discovery would be one whose importance could be compared to the discovery of quantization.

Of course, what he is trying to show may not be true, but that is a risk that any new fundamental idea in physics faces.

Ed Fredkin's style is not typical of scientists. He is more used to addressing boards of directors than lecture audiences. He learned the kind of physics that is in the Feynman lectures by spending time with Dick Feynman rather than reading his books. To some standard scientists, Fredkin at first seems like a nut. To be sure, some of his ideas are pretty nutty. But if you listen and think about it, there is much substance to what Fredkin has to say.

I gather that Fredkin has decided to spend some time around "ordinary physicists", to try and work out how his ideas fit in with current physical thinking. I believe you are very lucky that Fredkin wants to do this in your department.

Best wishes,
Stephen

And so it was that Ed became a research professor of physics at Boston University (BU). At MIT he'd gotten a DARPA grant that supported Tom Toffoli and Ed's only "physics PhD student" Norm Margolus in building ever-larger "cellular automaton machines". And when Ed moved to BU, this effort moved with him, leaving in effect "no trace of Ed" at MIT.

When Ed arrived at BU he found he was assigned to an office with a certain [Gerard 't Hooft](#)—who happens to be one of the more creative and productive theoretical physicists of the past half-century (and would win a Nobel Prize in 1999 for his efforts). Ed became friends with 't Hooft, inviting him and his family to spend time on his island, and later on the boat that Ed bought in the south of France. Feynman died in 1988, and Ed would tell me that he thought he'd "traded" one great physicist for another. (Feynman had suggested Ed try [Sidney Coleman](#), but Coleman wasn't into it.)

Like Feynman, I think 't Hooft felt a little uneasy with Ed's statements about physics. But in 2016 't Hooft ended up publishing a book entitled *[The Cellular Automaton Interpretation of Quantum Mechanics](#)*. I thought it was a nice recognition of 't Hooft's friendship with Ed. But Ed told me in no uncertain terms that he thought 't Hooft hadn't given him the credit he was due—though in reality I don't think what 't Hooft did was much related to Ed's actual work and ideas. (And, by the way, it's not directly related to my efforts either, though

conceivably looking at “generational states” in our Physics Project may give something at least somewhat analogous.)

In 1994 Ed’s direct affiliation with BU ended—though he remained on good terms with the department, and after I moved to the Boston area in 2002 I would often see him at an annual dinner the BU physics department put on for “Boston-area physics people”.

In 1998 Ed would summarize himself like this:

Ed Fredkin has worked with a number of companies in the computer field and has held academic positions at a number of universities. He is a computer programmer, a pilot, advisor to businesses and an amatuer [sic] physicist. His main interests concern digital computer like models of basic processes in physics.

For a while, Ed didn’t have a “university affiliation” (except, through Minsky, as a visitor at the MIT Media Lab), but in 2003—through his friend Raj Reddy—he became a professor (now of computer science) at Carnegie Mellon University, for a while spending time at their West Coast outpost, but mostly just making occasional trips in his plane to Pittsburgh.

Forty Years of Interactions with Ed

For a few years after I first met Ed in 1982, I’d see him fairly regularly. In 1983 I invited him to the first “modern” conference on cellular automata, that I co-organized at Los Alamos. I visited his house in Brookline, MA, a few times. I saw him at the Aspen Center for Physics, and at other places around the world. He was always fun and lively—and told great stories about all sorts of things. He gave the impression that he was mostly spending his time doing big things in business, and that science was an avocation for him. Sometimes he would talk about cellular automata—though I now realize that what he said was either very general and philosophical (leaving me to interpret things in my own way), or very specific to particular rules he’d engineered.

It was always a bit uncomfortable when it came to physics. Because the things Ed was saying always seemed to me pretty naive. Quite often I would challenge them—and frustratedly tell Ed that he should learn twentieth-century physics. But Ed would glide over it—and be off telling some other (engaging) story, or some such.

In 1986 I co-organized (with Tom Toffoli and Charles Bennett) a conference called *Cellular Automata ’86*—at MIT. Ed didn’t come—and I think I had the impression that he’d rather lost interest in cellular automata by that time. I myself went off to start my Center for

Complex Systems Research, and then to found Wolfram Research and start the development of Mathematica. Mathematica was released on June 23, 1988—and our records (yes, we’ve kept them!) show that Ed registered his first copy on December 14, 1988. In March 1991 I did a lecture tour about Mathematica 2.0, and saw Ed one last time before diving into work on my book *A New Kind of Science*—which led me for more than a decade to become an almost complete scientific hermit.

I saw Ed (now 62 years old) when I briefly “came up for air” in connection with the release of Mathematica 3.0 in 1996, and we continued occasionally to exchange pleasant emails:

Date: Sun, 29 Jun 1997 15:49:41 -0400

From: Ed Fredkin

To: Stephen Wolfram

...

[Reporting the birth of my second child]

...

For many children its worst when they are teenagers. Some glide through that period of life without hassle. Rick is doing great (at 15) despite his unorthodox education. He relishes calling his parents dopes, but aside from arguments about subjects like how late he should be able to hang out with his buddies, its clear that he doesn’t think we’re dopes.

...

I promise to read your book as soon as I get it!

...

Its nice to hear from you. News here is that I am no longer needed at Radnet as they now have a great CEO. I got a new airplane in December.

It’s called a Cessna CitationJet. It can carry 7 people at about 440 mph. So far its been a lot of fun. We’ll have to think of an excuse to go for a ride. We are planning to spend some time at Drake’s Anchorage in July. Its great for kids so if that interests you, let me or Joyce know.

I have taken as a challenge to architect a computer (that weighs a few kilos) that assumes another 100 years of Moore’s Law (10^{15} in cost

performance). There are a lot of unsuspected problems lurking in the details, but everyone of them seems to have easy solutions. I have given a number of talks (IBM Almaden and Watson labs, Intel, NYU, etc.). Interest in reversible computing has picked up since heat dissipation has gotten to be a really hot topic (no pun intended). The next high end Alpha may dissipate as much as 150 watts. Think of a light bulb!

I use Mathematica for something almost every week... keep it up!

Best regards,

Ed

Although I didn't see Ed myself for quite a few years, Ed would always write to ask for betas of new versions of Mathematica, and he would sometimes chat with staff from my company at trade shows. I thought it a bit odd in 1999 when I heard that in such an encounter he said that he was the one who had "introduced me to cellular automata". And, moreover, that he, Feynman and Murray [Gell-Mann] were the people who'd suggested I write SMP—which was particularly bizarre since, among other things, I hadn't met (or even heard of) Ed until about 3 years later.

Then, out of the blue on September 13, 2000, Ed calls my assistant, and follows up with an email:

Subject: Invitation

Date: Wed, 13 Sep 2000 23:53:09 -0400

From: Ed Fredkin

To: Stephen Wolfram

Hi,

The primary reason I'm contacting you has to do with a program I'm organizing at Carnegie Mellon (CMU). I wrote a proposal to the NSF, called

"The Digital Perspective" and got funded. The idea is to invite a number (8

to 10) of guests to come to CMU for a few days, to meet with students and to

give a Distinguished Lecture. The NSF would also like to arrange for the guests to come to Washington D.C. and give the same lecture there.

By “Digital Perspective” I mean looking at aspects of the world as Digital Processes. As you know, I am most interested in looking at physics this way. I have just started getting commitments from potential participants. Gerard ‘t Hooft has agreed to come and a number of other good physicists are thinking about it.

...

Please consider this to be a formal invitation. Of course, CMU will pay expenses and an honorarium. If the timing works out, it can probably be arranged for many of the students to have read your book before you come. You might get some good feedback from bright students who have also gained familiarity with the thoughts of others who are thinking about the “Digital Perspective”. The seminar will run throughout the 2000-2001 academic year.

If you can make it to CMU, I expect that it will be fun and interesting; both for you, for me and for many others.

...

I responded:

Subject: RE: Invitation
Date: Thu, 14 Sep 2000 06:49:19 -0500
From: Stephen Wolfram
To: Ed Fredkin

Thanks for the invitation, etc.

It sounds like a thing I’d like to do, but I can only consider *anything* after my book is finished.

...

If my book is done in time for your program, then, yes, I’d like to participate (though of course I’d want more details about the actual plans etc. etc.). But if the book isn’t done, then sadly I just can’t. If the cutoff time is June 2001, I am not extremely hopeful that the

book will be done ... but if it's fall 2001 the probabilities go up substantially (though, sadly, they are still not 100%).

...

And what are you up to these days? Business? Science? Other?

On another topic:

In my book, I'm trying very hard to write accurate history notes about the things I discuss. And for the notes on the history of cellular automata I've been meaning for ages to ask you some questions...

I'm not sure this is a complete list, but here are a few I've been curious about for a long time that I'd really like to know the answers to...

I know that history is hard ... even if it's about oneself. I consider that I have a good memory, but it's often hard for me to keep straight what happened when, and why, etc. But anything you can tell me about these questions ... or about other aspects of CA history ... I'd be very grateful for.

1. As far as you know, did you invent the 2D XOR CA rule? (I'm assuming the answer is "yes"...)
2. In what year did you first simulate this CA? On what computer? Where?
3. What other CA rules did you study at that time?
4. Do you still have any material from the simulations you did (printouts, tapes, programs, etc.)?
5. When you learn about the "munching squares" display hack? How did it relate to your work on the XOR CA?
6. What did you know about the work done by Unger etc. on cellular image processors? How did this relate to your work?
7. What did you know about von Neumann's work on cellular automata? How did it relate to your work?
8. What did you know about Ulam and others' work at Los Alamos on simulating cellular automata? How did it relate to your work?

9. Were you aware of work on cryptographic applications of CA-like systems?

...

Ed responded:

Subject: RE: Invitation

Date: Fri, 15 Sep 2000 01:25:23 -0400

From: Ed Fredkin

To: Stephen Wolfram

Hi,

Here are some answers and some free association type ramblings.

...

> And what are you up to these days? Business? Science? Other?

I'm winding down on business (I'm into one last e-business project) and like you, working on a book. My guess is that mine is nowhere as ambitious as yours... It's just to document my ideas about Digital Mechanics (Physics). In any case, these ideas have made more progress in the last 2 years than in the previous 40.

I bought a sailboat which is moored in Antibes, France. I spent most of the summer there and got more science done than in the prior several years. It's absolutely the perfect place and circumstance for me to work on my stuff. Gerard 't Hooft (plus wife and daughter) came down and joined us for a while. You know (I hope) about his interest in CA's? I'm going back there for a few weeks on Tuesday.

Here's a formal proof that you can, at any time, escape all your normal responsibilities and concentrate exclusively on one really important thing (hint, hint). The proof is that, at any time, YOU CAN DIE. I don't mean to be morbid, but sometimes it makes good sense to consider that proof

and
temporarily abandon all but some very important task (or some very
exciting
or fun thing).

Ed continued with a long response to my “history questionnaire”:

*> 1. As far as you know, did you invent the 2D XOR CA rule? (I'm assuming
> the answer is “yes”...)*

Yes, as far as I know I did invent it. Here is what I did. I decided to look for the simplest possible rule that met certain criteria. I wanted spatial symmetry and a symmetric rule vis-à-vis the states of the cells. The thought was to find something so simple that its behavior could be understood while not so simple as to be totally dull. The first such rule I tried was the XOR rule. I programmed it first on the PDP-1 (1961, at BBN and III) where I could see it on the display, and later I wrote a program for CTSS using a model 33 teletype as a terminal. My motivation was then, as it is now, to be able to capture more and more properties of physics within a Digital model. I found an easy proof as to why patterns reappeared in any number of dimensions. I also found, at the beginning, a formula for the number of ones as a function of time from a single one as the initial state. My recollection was that it was something like $2^D 2^{b(t)}$ where D is the number of dimensions, t is the time step, and $b(t)$ is the number of bits that are one in the binary representation of t (the tally function). After I showed all this to Seymour Papert, he generalized the proof re self replication from XOR (sum mod 2) to sum mod any prime. (Some time around 1967)

> 2. In what year did you first simulate this CA? On what computer?

Where?

See above.

> 3. What other CA rules did you study at that time?

I found a simple proof that a von Neumann neighborhood CA could exactly emulate any other (such as the 3×3 neighborhood) and used this as a reason to look at nothing else. I explored so many different rules that I probably would have found the game of Life had I not put blinders on. After I came to MIT (1968), I had 2 things in mind, to find a really simple Universal CA (I call them UCA's) and to find Reversible, Universal CA's (RUCA's) As you may know, the search for UCA's went slowly until I had the idea to abandon the Turing Machine model and look at modeling digital logic and wires. Within 15 minutes after this idea occurred to me, I had a 4 state UCA on my blackboard. At that time the best known was in Codd's thesis; an 8 state UCA. I showed this to a student of mine, Roger Banks, who had been struggling for a few years trying to complete an AI PhD thesis. The next morning both he and I showed up with 3 state UCA's. He switched his PhD topic and found a 2 state, von Neumann neighborhood UCA, a thing that Codd purported to have proved impossible.

While at BBN, after seeing all my 2-D CA's expanding with simple kaleidoscope like symmetries, (like the diamond shapes in the XOR rule), Marvin Minsky challenged me to find a rule (any rule) that showed spherical propagation. I took the challenge and shortly came up with such a rule.

With respect to reversibility, the first satisfactory RUCA was done by Norman Margolus. I shortly thereafter found a simple RUCA that didn't need the use of the Margolus Neighborhood trick.

*> 4. Do you still have any material from the simulations you did
> (printouts, tapes, programs, etc.)?*

Yes, Probably, quite a bit

*> 5. When you learn about the “munching squares” display hack? How
did it
> relate to your work on the XOR CA?*

I don't recall it having any effect. It's very unlikely that I knew of it prior to the XOR CA.

> 6. What did you know about the work done by Unger etc. on cellular image processors? How did this relate to your work?

I knew of it second hand, but I don't think it had any effect. Do you know about Farley and Clark (Wes Clark) and their publication while at MIT's Lincoln Labs in the late 50's?

> 7. What did you know about von Neumann's work on cellular automata? How did it relate to your work?

At the time I did the XOR work I had not read anything about the von Neumann CA, but I was told about it and I understood the concept very well. Many years later I read something (by Burkes, I think). I remember knowing that it was a 29 state system and that it knew left from right in order to extend and turn its construction arm.

> 8. What did you know about Ulam and others' work at Los Alamos on simulating cellular automata? How did it relate to your work?

All I knew about Ulam and CA is that, like the Hydrogen Bomb, he had key ideas but probably didn't get as much credit as he deserved. All my knowledge re Ulam was anecdotal. As to what he did vs. what von Neumann did I didn't really know anything. I didn't know anything about anyone else actually simulating CA's however I'm pretty sure I assumed that others must have done so. It was so easy and so obvious. While the use of a computer with a display (such as the Lincoln Lab TX-0 and TX-2, the Digital PDP-1 and the IBM 709 and 7090 all had or could have CRT displays, it was easy enough to display simple CA's with a printer, even a 10 CPS teletype.

> 9. *Were you aware of work on cryptographic applications of CA-like
> systems?*

I thought I invented that idea! As soon as I found ways to make RUCA's
it

occurred to me that they could be used for cryptography. As an aside,
when

Witt Diffey [Whit Diffie] came up with the idea of public key
cryptography,

which needed a trapdoor function, I thought of using the product of 2
large primes.

I had just written the first program, in LISP, to implement Michael
Rabin's first

version of a probabilistic prime test. As soon as I implemented it I
started a search at 10^{100} and discovered that $10^{100} + 35,737$ and
 10^{100}

$+ 35,739$ were prime. A week later I met Rich Schroepel in LA (he was
working for my company, Ill) and knowing a larger prime pair than
anyone

else on Earth I told Rich and he was blown away. He was seated at a
PDP-10

terminal and all he said was an emphatic "Really!" He then went type,
type,

type for a few seconds and turned around and said "You're right!"
which blew

me away! I asked what he did and he said (while knowing nothing of
Rabin's

method) "all I did was look at $3^{(n-1)} \bmod n$, you know, Fermat's little
theorem, it usually gives 1 for primes."

I'm rambling, probably about stuff of no interest to you. Anyway, I
stopped

Ron Rivest in the hallway at Tech Square and asked if he had heard of
Diffey's [Diffie's] stuff. I don't remember exactly what he said but I
know that when

I told him that Rabin's new method to find large primes meant that the
product of 2 primes was a good trapdoor function he was surprised
and

thought it was a good idea! I never thought any more about it and
hadn't

come up with the idea of using the phi function... Years later, long after
RSA was a big thing Ron reminded me of the event... Don Knuth told
me that

he also thought of using the product of 2 primes before RSA, but he
couldn't

have known about Rabin's method when I did (as Rabin told it to me

right
when he thought it up!)

By the way, I have an interesting algorithm for factoring smaller numbers, such as can be done in less than an hour with Mathematica (normal FactorInteger or ECM). I've written a few terribly unoptimized Mathematica functions that implement the method. For what its good for, my Mathematica functions (not compiled or anything) make Mathematica factor in a lot less real time than Mathematica does with FactorInteger or ECM.

The big news re me and my work is what's happening right now. Whatever one thinks about my stuff (Digital Mechanics), it's vastly improved. However it's still very far from a complete theory. Of course, Digital Mechanics is about CA's.

If you have any interest in reversibility, I've done lots in that area, ranging from RUCAs, conservative logic, and my transforms. The transforms are general methods of converting algorithms that calculate the approximate time evolution of a system (approximate because of round off, truncation and the finite Δt) which is approximately reversible (by changing Δt to minus Δt) into an equivalent algorithm that calculates approximately the same thing going forwards, but which is exactly reversible (being calculated on a computer with round off and truncation error). I also have a lot of methods for making RUCA's with particular properties.

You've criticized me in the past for not publishing stuff, but I'm so ambitious as to what I'm trying to do that I haven't had the motivation to publish all the little things I've uncovered along the way.

I'm sure I discovered more and better ways to make all kinds of RUCAs before anyone else with the exception of the rule found by my student, Margolus.

Finally, one last anecdote. You and I were at some meeting long ago (maybe

Santa Fe?) and you brought along an early Sun to demonstrate your collection

of different kinds of 1-D CA's. After your talk, I asked you why none of the CA's you showed were reversible. Your response was "Because all reversible CA's are trivial." That really was a very common belief, coincident with most people's intuition. On the spot, I made up a rule, using your convention for specifying it, of an "interesting" reversible CA.

You typed it in and ran it. Being surprised is one of the best kinds of experiences we ever have.

As Emerson once quipped, "My apologies for such a long email, I didn't have the time to write you a short one."

I'm having fun; it's a good thing to do!

Best regards

Ed F

PS If you have any interest in having parts of your book read so that you can get comments prior to publication, I have an idea that might be useful.

A little later he added:

Subject: error

Date: Fri, 15 Sep 2000 10:12:15 -0400

From: Ed Fredkin

To: Stephen Wolfram

Hi,

Looking at my long email I noticed a boo boo.

Where I wrote, quoting Schroeppele talking about $3^{(n-1)} \bmod n$, "...it usually give a 1 for primes..." very true but a bit of an understatement. Of course, it ALWAYS gives a 1 for primes! What Schroeppele said was that it

usually doesn't give a 1 for non-primes. It's incorrect for 91 and 121 and

lots of other small numbers, but seems to work better for large numbers...

but then you probably know much more about such things than I do.

Also

looking at your questions, I had the feeling that some might have been prompted by my circa 1990 Digital Mechanics paper. If so, I guess I repeated stuff already in the paper and I apologise.

Regards,

Ed F

I responded, asking for various pieces of clarification (and now that I'm writing this piece I would have asked even more, because some key parts of what Ed said I now realize don't add up):

Subject: Re: your mail
Date: Wed, 20 Sep 2000 21:04:54 -0500
From: Stephen Wolfram
To: Ed Fredkin

...

>> 1. As far as you know, did you invent the 2D XOR CA rule?

>>

> Yes, as far as I know I did invent it. Here is what I did.

>

Very interesting.

1a. Did you ever look at 1D CAs? If not, why not?

1b. Did you think about analogies between XOR rules and linear feedback shift registers?

1c. Did you think about analogies between XOR rules and Pascal's triangle?

By the way, the result about the number of binomial coefficients mod a prime has been independently discovered a remarkable number of times

(including by me). The earliest references I know are Edouard Lucas (1877) and James Glaisher (1899).

....

>> 3. What other CA rules did you study at that time?

> ... I explored so many different rules that I probably would have found the game of Life had I not put blinders on.

By the way, I happened to have a long phone conversation recently with

John Conway about the history of the Game of Life. I still haven't quite got to the bottom of exactly what Conway was doing and why (I think he wants some of the history lost, which is a pity, because it is interesting and reflects much better on him than he seems to believe...) But what is clear is that Conway (and his various helpers) had much more serious motivations from recursive function theory etc.

than is ever usually mentioned. It was just not a "find an amusing game" etc. piece of work.

> Marvin Minsky challenged me to find a rule (any rule) that showed spherical propagation. I took the challenge and shortly came up with such a rule.

I don't believe I've ever seen your rule of this kind. I showed such a rule to Marvin in 1984 and he said "that's very interesting; we were looking for these but hadn't found any". So I'm confused about this....

...

>> 6. *What did you know about the work done by Unger etc. on cellular image processors? How did this relate to your work?*

> I knew of it second hand, but I don't think it had any effect.

Wasn't BBN quite involved with cellular image processing? And I believe

you worked on aerial photography analysis. Did you use cellular automata for image processing?

...

>> 9. *Were you aware of work on cryptographic applications of CA-like systems?*

> I thought I invented that idea!

There was a lot of work done on 1D CAs by some distinguished mathematicians consulting for the NSA in the late 1950s. I think much of it is still classified. But over the years I've talked to many of the people involved (Gustav Hedlund, Andrew Gleason, John Milnor, some

NSA folk, etc. etc.), and read their unclassified papers. They figured out some interesting stuff. They thought of it as related to nonlinear feedback shift registers.

> As soon as I found ways to make RUCA's it occurred to me that they could be used for cryptography.

How?

There's a 1D CA (rule 30) that I studied in 1984 that has been extensively used as a randomness generator (e.g. Random[Integer] in Mathematica uses it), and that has been used a bit as a cryptosystem.

I tried to make a good public key system out of CAs in the mid-1980s (mostly in collaboration with John Milnor), but did not come up with anything satisfactory. ...

...

> I also have a lot of methods for making RUCA's with particular properties.

I am definitely somewhat interested in these things. They don't happen to be central to my grand scheme. But they are obviously worthwhile

...

AND WORTH (you) WRITING DOWN!!

I'm sure I discovered more and better ways to make all kinds of RUCAs before anyone else with the exception of the rule found by my student, Margolus.

Interesting. You probably know that the general problem of telling whether an arbitrary 2D CA is reversible is undecidable (the question can be mapped to the tiling problem).

So I'm taking it that you have some good methods for generating 2D reversible CAs. That's obviously interesting.

> Finally, one last anecdote. ... I asked you why none of the CA's you showed were reversible. Your response was "Because all reversible CA's are trivial." ...

This anecdote can't be quite right. I have known since 1982 that there are nontrivial things that can happen in CAs that are made reversible by your mod 2 trick. What is true (and may have been what I was saying) is that none of the 2-color nearest neighbor CAs that are reversible are non-trivial. With more colors or more neighbors, that changes. I'm

guessing that what you showed me was a 4-color nearest neighbor CA that is reversible ... and that is of course quite easy to get by recoding a 2-color one that has your mod 2 trick.

By the way, I heard third hand a while back that you had “introduced me to CAs”. For what it’s worth, that isn’t correct. My first “CA experience” was actually in 1973 (when I was 13) when I tried to program molecular dynamics on a very small computer, and ended up with something equivalent to the square CA fluid model. My next CA experience was in summer 1981. I was trying to make models of “self organizing” systems (now I hate that term), particularly self-gravitating gases. I ended up simplifying the models until I got 1D CAs. That fall I spent a month at the Institute for Advanced Study, and spent a lot of time studying von Neumann’s work, etc., and analysing all sorts of features of 1D CAs. I came for a day to give a talk at MIT, and was having dinner with some LCS people (Rich Zippel was one of them), and they told me about your work. Later that fall I talked with Feynman a certain amount about what I was doing with CAs, and he again mentioned you. (I think he had been to your Physics of Computation meeting, which was perhaps in June 1981, but I didn’t discuss the CA aspects of the meeting with him.) Then in [January 1982] I came to the meeting you had on your island, and Tom Toffoli showed me his 2D CA machine (at the time he gave me the impression of 95% hackery, 5% science), and you showed me the 2D XOR CA on a PERQ computer.

Ed didn’t respond to this, but three days later we talked on the phone. I sent some (unvarnished) notes from the call to a research assistant of mine:

Subject: Fredkin conversation
Date: Sat, 23 Sep 2000 03:05:43 -0500
From: Stephen Wolfram

I had a long conversation with Ed this evening.

About his work in science, my work in science, etc.

A few things mentioned:

- He feels bitter that his paper on reversible logic, coauthored with Tom Toffoli, was actually all his (Ed's) work
- He is pleased that I will discuss history even when people haven't published things (of course he has published little)
- He says he has written about 150 pages about his views of physics; he is planning to prepare something, perhaps for publication, in about a year
- He says he missed not being able to bounce ideas off Dick Feynman
...
even though Feynman often ended up screaming at him (Ed) about how dumb his ideas were
- He said that his main problem was that he has been trying to get people to steal his ideas for years, but nobody was interested
- He said that now "for some reason" he is becoming more concerned about matters of credit
- He is a serious fan of Mathematica, the Mathematica Book, etc.
- He made an effort again (he's been trying for 20 years) to get me to coauthor a paper with him. He recognizes that he can't write a credible scientific paper, but he's "sure he has some ideas I haven't thought of". I told him that unfortunately I haven't written a paper for 15 years.
- I told him that particularly when I'm in the Boston area, I'll look forward to chatting with him about physics etc.
- He said he's tried to interact some with Gerard 't Hooft, but that 't Hooft keeps on rushing off in traditional physics directions that Ed (and I, by the way) think are stupid
- He wanted to know if I really believed that all of physics etc. was ultimately discrete; he expressed the opinion that he and I may be the only people in the world who actually believe that right now
- He told a bizarre story about how Don Knuth gave a talk at MIT recently on computers and religion, and how 1/4 of it was stuff that Don had heard about from Ed. Apparently Guy Steele asked a question

about

how Don's stuff related to Ed's, and Don said something meaningless.

I talked to him a little more about the CA history stuff. He mentioned that around 1961 a certain Henry Stommel (sp?) told him that CA-like models had been used in studying sand dunes in the 1930s. I have a feeling this may be another cat gut search, but perhaps we can follow up. (You could email Ed at the appropriate time.)

I asked Ed if he had ever looked at cryptography (as in NSA style stuff) with CAs. He said no. But that in the late 1960's he had had a student who had studied ways to make counters out of JK flip flops ... and that that person's work had made something that Ed thought could be used for cryptography. This was followed up by a certain Vera Pless subsequently.

I didn't hear anything more from Ed for a while, though a public records search indicates that, yes, he had successfully "worked the system" to [get \\$100k from the NSF for "The Digital Perspective Project"](#). And on May 1, 2001, I received a rather formal email from Ed (for some reason Americans born before about 1955 seem to reflexively call me "Steve"):

Subject: Workshop on the Digital Perspective 24-26 July, Washington DC

Date: Tue, 1 May 2001 21:20:45 -0400

From: Ed Fredkin

To: Steve Wolfram

We are sending this email to invite you to an NSF-sponsored workshop on the

Digital Perspective in Physics planned for July 24th through the 26th, Tuesday, Wednesday and Thursday. It will be held in the NSF building, Arlington Virginia. Gerard 't Hooft has already agreed to present a paper

and we hope that you will also be willing to contribute. We intend to combine the papers presented at the workshop into a monograph that will be

published later this year. Two earlier workshops on related subjects were

held at Moskito Island and this was a central theme at a meeting held at

MIT's Endicott house in 1982. Participants at previous meetings included

Charles Bennett, Richard Feynman, Ed Fredkin, Leo Kadanoff, Rolf Landauer,

Norman Margolus, Tomasso Toffoli, John Wheeler, Ken Wilson,
Stephen Wolfram
and others.

...

When I didn't immediately respond, Ed called my assistant, saying that he was "calling regarding a meeting he spoke with [me] about on the phone". I responded by email later the same day:

Subject: I gather you called...

Date: Tue, 15 May 2001 15:18:57 -0500

From: Stephen Wolfram

To: Ed Fredkin

Sorry for not getting back to you sooner....

I myself am right now trying to work at absolutely full capacity to finish my book/project. I haven't done any travelling at all for a long time, and won't until my book is done.

And I also don't yet have anything public to say about my work on physics.

Hopefully by the end of the year my book will be done and I will have quite a bit to say.

However, it occurs to me that one or two of my assistants might be very good people to come to your workshop.

Who all is coming?

One person you should definitely invite is someone who has been an assistant of mine, and now works part time for me, and part time on his own projects. His name is David Hillman, and he's been interested in discrete models of spacetime for a long time. (He got his PhD working on some kind of generalization of cellular automata intended as a spacetime model.)

I have two physics assistants, and one math one, who might be relevant for your workshop.

Just let me know in more detail who might be coming, and I'll try to figure out the correct person/people to suggest.

Of course I'd love to come myself if I were a free man. But not until the book is done.

In haste,

— Stephen

Ed responded pleasantly enough:

Subject: RE: I gather you called...
Date: Tue, 15 May 2001 17:08:39 -0400
From: Ed Fredkin
To: Stephen Wolfram

Hi,

Sorry you can't make it.

About half of those coming are veterans of some Moskito Island workshop.

Newcomers include Gerry Sussman, Tom Knight, Gerard 't Hooft, John Negele, John Conway, Raj Reddy, Jack Wisdom, Seth Lloyd, David di Vincenzo, plus a number of students, etc. A couple of those mentioned are still struggling with scheduling issues.

But, in any case, I would be pleased to have David Hillman come to the workshop. Send me his email address and I will send him an invitation.

Best regards and good luck on the book!

Ed

I responded and suggested an additional person from our team for his workshop. Nearly a month passed with no word from Ed, so I pinged him asking what was going on. No response. It was a very busy time for me, and this wasn't something I wanted to be chasing (I saw myself as doing Ed a favor by suggesting sending people to his workshop) ... so I sent a slightly exasperated email:

Subject: your conference, again

Date: Fri, 15 Jun 2001 06:08:39 -0500

From: Stephen Wolfram

To: Ed Fredkin

Look ... I'm now in a bit of an embarrassing situation: following your initial response, I told David Hillman and David Reiss about your conference

... assuming you'd want to invite them ... and they both became quite interested in it. But they never heard from anyone about it. So now of course they're wondering what's going on. And so am I. What should I tell them? I'm now embarrassed about having suggested this...

This seems peculiarly un-you-like. I was thinking you must have been away or something. But isn't the conference coming up very soon?

I hope everything's OK...

Still no response from Ed. A week later I called him, and we talked for two hours. It wasn't clear why he hadn't already reached out to the people I'd suggested, but he quickly said he would. And then Ed launched into telling me about the “astounding” cellular automaton models he said he'd just created that “had charge, energy, momentum, angular momentum, etc.”. He talked about things like the idea of what he called an “infoton” that would be an “information particle” that would “make Feynman diagrams reversible”. I explained why that didn't make any real sense given how Feynman diagrams actually work. It was the same kind of conversation I had many times with Ed. I kept trying to explain what was known in physics, and he kept on coming back with things that, yes, I think I understood, but that seemed close to typical crackpot fare to me. But Ed seemed convinced he had discovered something great (though exactly what I couldn't divine). And eventually—having

obviously not convinced me of what he was doing “on its merits”—he just came out and said “It must be related to stuff you’re doing, one way or another”.

I explained that I really didn’t think that was very likely, not least because I emphatically wasn’t trying to use cellular automata as models of fundamental physics. And with that, Ed launched into a long speech about giving credit, particularly to him. I explained that I was trying hard to write correct history, and reiterated some of the questions I’d asked him before. He didn’t really tell me more, but instead regaled me with stories (that I’d mostly heard many times before from him) about how he’d been the first to figure out this and that—apparently oblivious to historical research I tried to tell him. But eventually we both had to go—and the conversation ended pleasantly enough, with him confirming the email addresses for the two people for his workshop.

As the workshop approached, the people from my team had made arrangements to go to Washington, DC—but still didn’t know where exactly the workshop was. With days to go, one of them simply called Ed to ask. But Ed told them that actually they couldn’t come, because “Raj Reddy says there is no room for you”. Really? No extra chair to be found? Ed was the organizer, wasn’t he? Why was he laying this on someone else? It seemed to me that Ed was playing some kind of game. But at that moment I was too busy trying to finish my book to think about it. (Now that I’m writing this piece, however, I realize that Ed was perhaps following an “algorithm” he’d established years earlier when he was proud to have organized a meeting to push forward his ideas about timesharing—by inviting just people who supported his ideas, and not inviting ones who didn’t. I don’t know if the meeting actually happened, or what went on there. I don’t think the writeup promised in the invitation and in the NSF contract ever materialized.)

In January 2002 *A New Kind of Science* was off to the printer, and review copies were starting to be sent out. In late March a seasoned journalist named Steven Levy (who had written about my work on cellular automata in the mid-1980s) was talking to someone from my team and reported that Ed had told him that “Minsky had told [Ed] to publish his stuff on the web to stake out priority” before my book came out. (And it’s a pity Ed didn’t do that, because it might have made it clear to him and everyone else how different what he was saying was from what I was saying.) But in any case Levy said that Ed seemed to be saying the same things as he’d said 15 years ago—and Levy knew that regardless of anything else I’d done incredibly much more since then.

After his conversation with Levy, Ed sent me mail:

Subject: The Book

Date: Fri, 22 Mar 2002 16:38:29 -0800 (PST)

From: Ed Fredkin

To: Stephen Wolfram

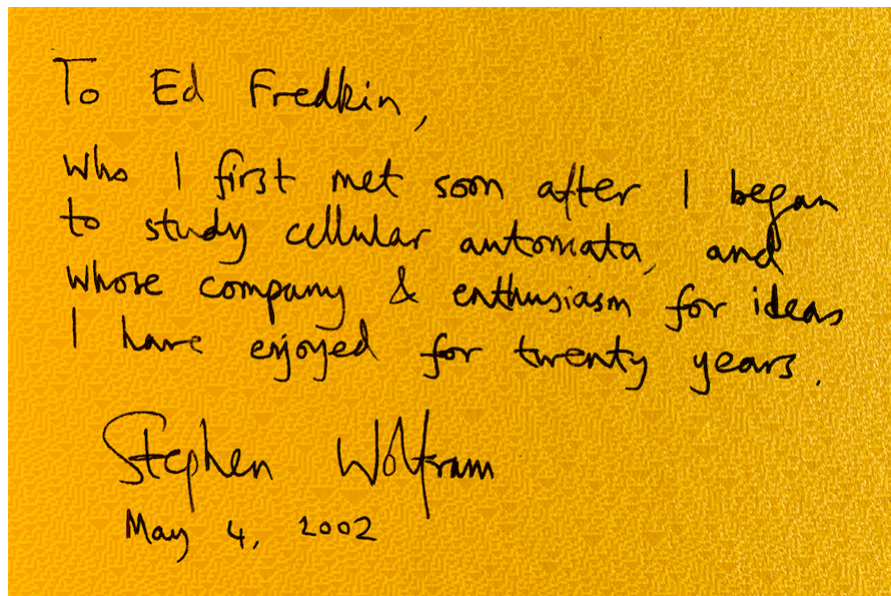
Congratulations on finishing!!!

I ordered the book from someplace, so long ago I can't remember from who. I'm wondering if, when its possible, I could get a copy in advance of whenever my ordered copy is going to appear. I just don't want to be the last on the block to see it. Of course I'd be happy to pay if you can tell me how to do it.

Thanks,

Ed Fredkin

The book was going to be published on May 14; on May 4 I signed a copy for Ed:



To Ed Fredkin,
who I first met soon after I began
to study cellular automata, and
whose company & enthusiasm for ideas
I have enjoyed for twenty years.
Stephen Wolfram
May 4, 2002

The book mentioned Ed a total of 7 times. (The person with the most mentions overall was Alan Turing, at 19; Minsky had 13; Feynman 10.)

Ed never told me he'd received the book. And I'm not sure he ever seriously looked at it. But somehow he was convinced that since he knew it talked a lot about cellular automata, and had a section about physics, it must be about his big idea—that the universe is a cellular

automaton. As one witty friend pointed out to me in connection with writing this piece, my book says only one thing about the universe being a cellular automaton: that it isn't! But in any case, Ed apparently seemed to feel that I was stealing credit from him for his big idea—and, as I now realize, started an urgent campaign to right the perceived wrong, basically by telling people that somehow (despite all my efforts to describe the history) I wasn't giving anyone enough credit and that "he was there first". The *New York Times* rather diplomatically quoted Ed as saying "For me this is a great event. Wolfram is the first significant person to believe in this stuff. I've been very lonely". It followed up by saying that "Mr. Fredkin, who said he was a longtime friend, said Dr. Wolfram had 'an egregious lack of humility'". (In some contexts, I suppose that might be a compliment.)

In writing this piece I asked Steven Levy what Ed had actually said in the interview he did. His first summary in reviewing his notes was "He says he considers you a friend and then goes on endlessly about what an egomaniac you are". But then he sent me his actual notes, and they're somewhat revealing. Ed doesn't claim he introduced me to cellular automata, perhaps because he realizes that Levy knows from the 1980s that that isn't true. But then Ed tells the story about showing me reversible cellular automata, which I'd explained to Ed wasn't true. Ed goes on to say that "Everyone who's in science wants credit, driven probably by wanting to become famous. [Wolfram] has a larger than normal dose". Ed says that when he had said that cellular automata underlie physics, I'd said that was crazy. (Yup, that's true.) But then Ed said "Now he denies this". Huh? Ed went on: "He's a prisoner of some kind of overactive ego. I believe he might not know. Wolfram deserves loads & loads of credit, but he has this personality flaw". And so on.

A month later Ed writes to me:

From: Ed Fredkin
To: Steve Wolfram
Sent: Friday, June 14, 2002 2:48 PM
Subject: ANKOS critics

Steve,

Sometime soon I'd like to get together and talk.

I've read a lot of your book.

Take a look at the draft of a little paper of mine (attached). I'd appreciate comments.

Ed F

The following is my response someone else's response [Gerry Sussman] to a review of ANKOS.

My comments are only with regard to Wolfram's ideas on modeling physics.

I don't happen to like his network model but we are in agreement that some kind of discrete process might underlie QM.

Not everything Wolfram says is wrong.

...

The ideas that some kinds of discrete space-time processes (such as CA's) might underlie physics or other processes in nature is the BABY. Everything else in ANKOS (or missing from ANKOS) is the BATH WATER.

Ed's attached paper was basically yet another restatement of cellular automata as models of fundamental physics.

A few weeks later there was a strange (if in some ways charming) incident when a reporter for the *San Francisco Chronicle* decided to investigate what seemed to be a science feud between Ed and me. After a nod to medieval metaphysicians, [the article \(under the title "Cosmic Computer"\)](#) opens with "Nowadays, with a daring that might have dazzled St. Augustine and St. Thomas Aquinas, two titans of the computer world argue that everything in the universe is a kind of computer." After analogizing me to Britney Spears, the article goes on to say "The excitement has also brought tension to the long-standing friendship between Wolfram and Fredkin, who are now wrestling with one of the bigger bummers of any scientist's life: a dispute over originality." The article reports: "Last week, the two men had a long, heartfelt phone conversation with each other, in which they tried to resolve their strong disagreement over priority. The conversation was amicable, but they failed to reach agreement."

And so things remained until March 2003 when Ed sent the following:

Subject: Re: NKS 2003 Conference & Minicourse
Date: Thu, 20 Mar 2003 17:24:59 -0500
From: Edward Fredkin
To: Stephen Wolfram

Dear Stephen,

I guess I'm on a Wolfram mailing list for potential attendees for your Boston conference. I hope you don't mind a little plain speaking. I consider that I am a friend of yours and therefore I take the risk of telling the emperor about his new clothes. Of course, few others would do so as a friend. Please don't be offended as the plain talking that follows is my attempt at trying to be constructive.

Your work is acquiring a reputation amongst the scientific community that is much less than it deserves. I find myself often in the position of defending you, your work and your accomplishments against

the negative views that many hold, even though they have little understanding of the significance of what you have done. They are turned off by your egregious behavior; it distracts much of the scientific community rendering it barely possible for them to take you seriously. You have invented and discovered quite a few things, but so have others. You told me you would try to give credit in ANKOS where credit was due; I believed you and I believe that you tried your best but nevertheless you failed miserably. I guess you simply didn't know how. Consider this conference: Must this conference be a one man

show or might it actually be better for the ideas in ANKOS and better for SW and his overall scientific reputation if it were a real conference where others might address the same questions? Please don't

kid yourself into thinking that no one else has anything original, novel, important or interesting to say.

Of course, this so-called "...first ever conference" devoted to the "...ideas and implications..." of concepts found in ANKOS might be nothing

more than a marketing tool for Mathematica and for sales of the ANKOS

book. If so, you ought to call a spade "spade".

You've done enough things (and hopefully will continue to do so) to ensure your reputation as a pioneer in various areas. This flood of self puffery simply detracts, in the minds of many whose opinions you ought to value, from the positive reputation you deserve.

I'm not one of those whose opinion of your work is in any way affected by your unfortunate behavior. I see and understand exactly what you've

done and I know and understand what your work is based on. I am human,

so I find it interesting when you now and then claim to have discovered

an idea or fact that I personally explained to you when it was perfectly clear at the time that to you, the idea was absolutely novel. My model of you is that your overpowering motivation results in your mind playing tricks on you. I really believe that you actually forget; that you actually re-remember the past differently than it happened.

But I am the eternal optimist. I believe that even Stephen Wolfram might someday come around and join the collegial scientific community where you receive credit and give credit; both nearly effortlessly. The world actually might voluntarily heap honors on you as opposed to SW having to orchestrate “conferences” for the glorification of SW and all the ideas claimed by SW. No one knows better than me how slow and torturous this process can be for new and novel big concepts, but patience and modestly [*sic*] still seems like the better path.

Please try to not be offended. I actually mean well. If you ever have an actual, real conference, invite me to be a speaker; I’ll come. If I organize another conference you can rest assured that you will be invited again (as you were for the NSF Workshop) and I hope you will come to talk about your ideas and maybe — maybe even stay to hear what others have to say on the subject. It’s not too healthy to the scientific mind to be the only real speaker at conferences you organize and hype for yourself.

Among the very few who really are able to appreciate what you’ve done, I am one of your greatest supporters. But I am not your average person with more or less normal reactions. When you reach for extra glory and credit by stealing one of my ideas, my reaction is: “I admire your good taste”.

Best regards

Ed F

On Thursday, March 20, 2003, at 02:19 PM, Stephen Wolfram wrote:

> *In June of this year we’re going to be holding the first-ever*
 > *conference devoted to the ideas and implications of A NEW KIND OF*
 > *SCIENCE. I think it’ll be an exciting and unique event. And if you’re*
 > *interested in any facet of NKS or its implications, you should plan*
 > *to come!*
 >
 > *I’ll be giving a series of in-depth lectures to explain the core*

> ideas of NKS. There'll be more specialized sessions exploring
 > implications and applications in areas such as computer science,
 > biology, social science, physics, mathematics, philosophy, and future
 > of technology. And there'll also be workshops and case studies about
 > such issues as modelling, computer experimentation, defining NKS
 > problems, NKS-based education—as well as a gallery of NKS-based
 > art pieces.

>

> I'd expected that it'd be a few years before it would make sense to
 > start having NKS conferences. But things have gone faster than I
 > expected, and the enthusiasm and energy we've seen in the ten
 months

> since the book was published has made it clear that it's time to have
 > the first NKS conference.

>

> In planning NKS 2003, we want to cater to as broad a range of
 > attendees as possible. There'll be many professional scientists
 > coming, as well as technologists and other researchers from a very
 > wide range of fields. There'll also be a large number of educators
 > and students, as well as all sorts of individuals with general
 > interests in the ideas and implications of NKS.

>

> We'll be holding NKS 2003 near Boston over the weekend of June 27-29,
 > 2003. There's more information and registration details at
 > <http://www.wolframscience.com/nks2003>

>

> It's going to be an extremely stimulating weekend—and a unique
 > opportunity to meet a broad cross-section of people interested in new
 > ideas.

>

> I hope you'll be able to be part of this pioneering event!

>

>

> — Stephen Wolfram

I responded:

Subject: Re: NKS 2003 Conference & Minicourse
Date: Sat, 22 Mar 2003 22:19:33 -0500
From: Stephen Wolfram
To: Edward Fredkin

Ed —

I must say that I am reluctant to respond to a note like the one below, but it seems a pity to let things end this way.

I can tell you're very angry ... but beyond that I really can't tell too much.

I'd always thought we had a fine, largely social, relationship. We talked about many kinds of things. It was fun. Occasionally we talked about science. In the early 1980s I learned a few things about cellular automata from you. None were extremely influential to me, but they were fine things that you should be proud of having figured out—and in fact I took some trouble to mention them in the notes to NKS.

You also told me some of your thinking about fundamental physics. I was (I hope) polite, and tried to be helpful. But I always found what you were saying quite vague and slippery—and when it became definite it usually seemed very naive. I think it's a great pity that you've never taken the time to learn the technical details of physics as it's currently practiced. There's a lot known. And if you understood it, I think you'd be able to tell quite quickly which of your ideas are totally naive, and which might actually be interesting.

I think it's also a pity that—so far as I can tell—you've never really taken the time to understand what I've done. It's in the end pretty nontrivial stuff. It's not just saying something like “the universe is a cellular automaton” or “I have a philosophy that the universe is like a computer”. It's a big and rich intellectual structure, built on a lot of solid results and detailed, careful, analysis. That among other things happens to give a bunch of ideas about how physics might actually work—that have (so far as I know) almost nothing to do with things you've been talking about.

I do agree with your belief that the universe is ultimately discrete. But of course many people have for a long time said that they thought the universe might at some level be discrete. Some of those people (like Wheeler, Penrose, Finkelstein, etc.) are sophisticated physicists, and what they've said has lots of real content—it's not just vague essay-type stuff. Now, I don't happen to think what they've specifically proposed is correct. But you would be completely wrong to think (as you seem to)

that
somehow the idea that the universe might be discrete originated with
you.

I really encourage you to read NKS in detail, including the notes at the
back. I think there's a lot more there than you imagine. And I think if
you really understood it, you would be completely embarrassed to
write a
note like the one below.

You've never struck me as being someone who is terribly interested in
other
peoples' ideas. And that's of course fine. But you shouldn't assume you
know their ideas just on the basis of a few buzzphrases or some such.
In
some areas of business, that approach often works. Because, as we
both
know, the ideas typically aren't that deep. But it won't work with a
character like me doing science. There's too much nontrivial content.
You
have to actually dig in to understand it. And from the things you say
you
obviously haven't.

For twenty years I thought we had a fine personal relationship. I
thought
it was a little odd that you seemed to go around telling people that you
had
introduced me to cellular automata. We talked about this a few times,
and
you admitted this wasn't a true story. But while I thought it was a little
unreasonable for you to keep on saying something you knew wasn't
true, I
didn't pay much attention. It never really got in the way of our
relationship.

And then there was the incident of your NSF-funded conference. You
invited
me. I said I couldn't come. And suggested two alternates. You said fine.
But then you never contacted these people. Which was rather
embarrassing
for me. And then, when David Reiss contacted you, you told him the
conference "was full".

Later, when we talked about it, you admitted that that was a lie—and
then
blamed the lie on Raj Reddy.

Frankly, I was flabbergasted by all this. That's not the kind of interaction someone like me expects to have with a seasoned high-level

operative like yourself. Yes, that's the kind of thing some sleazy young businessperson might do. But not a mature businessperson who has run companies and things.

I still have no idea what you were thinking of. But it thoroughly shook my confidence in you as someone I could interact straightforwardly with.

And then, of course, there's the question of what you've said to journalists

etc. about NKS. In detail, I don't have much idea. But something fishy was surely going on. I haven't gone and studied all the quotes from you.

But certainly my impression was that you were trying to claim that really lots of key things in NKS were things you had done or said first.

You know that I tried to research the history carefully. And unless I missed something quite huge, your contributions to NKS were extremely minor, and are certainly accurately represented in the history notes. Now of course if you don't actually understand what I've done in NKS, that may be hard to see. But I can't really help you with that.

OK, where do we go from here?

We talked at some length when that reporter from a San Francisco paper was trying to write a story about you and NKS. I thought we had a decent conversation. But then, so far as I could tell, you went right ahead and told the reporter—again—exactly a bunch of things we'd agreed in our conversation weren't true.

It was the same pattern as with telling people that you'd introduced me to cellular automata. And it resonated in a bad way with the lie you told about your conference.

I would have expected vastly better from you. I must say that I was personally most disappointed. And I concluded with much regret that I must have seriously misjudged you all these years.

I would like nothing more than to be able to mend our relationship,
and go
back to the kind of pleasant social interactions we have always had.

How can that be achieved? Perhaps it's impossible. But one step is that
you might actually try to understand what I've done in NKS. That
would
surely help.

— Stephen

Subject: Delayed reply
Date: Thu, 3 Apr 2003 23:34:17 -0500
From: Edward Fredkin
To: Stephen Wolfram

Stephen,

I have been traveling and more recently have had my time gobbled up
by
a most urgent matter.

I appreciate your quick reply to my email and I will get back to you
sometime soon. Rather than trying to respond to everything you
brought up,
I will be limited to dealing with a couple of issues at a time.

What I can tell you is that I am not angry, and was not angry or upset.
I have always been a non-emotional observer with regard to whatever
it is
that comes my way. That's just my nature. It has come in handy at
times,
such as when someone's stupid mistake caused the single engine jet
fighter
I was flying have an engine fire on take off. This required shutting down
the
engine and taking other drastic actions very quickly; no time to get
mad.

The gist of my comments to you was not related to the work you
documented in NKS, but rather to the style and methodology you are
using
while trying to get people to understand and appreciate what it is you
have done. I certainly agree with the fact that it is extraordinarily
difficult to get the scientific establishment to pay attention, listen,
understand and appreciate what you've done. Nevertheless, I think

there
might be a better approach to that problem than the one you are
following.

So, as soon as I can get a little breathing room, I'll respond to some
of your comments. I do value our friendship and whatever I do in this
regard
will be an attempt at honest and unemotional communication with the
goal of
some better mutual understanding.

By the way, I have taken the time to read and understand what you've
done in NKS. I'm pretty sure that I am better able than most to
appreciate
the effort, persistence and creativity that went into that work.

You have made some comments about me and my own work, and I
wonder what
you actually know about it beyond our conversations and the things
you
referenced in NKS.

As soon as I can get some time I'll continue with some further
thoughts.

Best regards,

Ed

Ed didn't send the promised followup. But a couple of months later *New Scientist* sent our
media email address a note titled: "cover feature on Fredkin, Wolfram right to reply", which
asked for "comment on the suggestion that you first became familiar with cellular automata
first at Fredkin's lab in the 1970s and that examples in *A New Kind of Science* came out of
work done in the lab". I told Ed he should correct that—and he responded to me:

Subject: Re: cover feature on Fredkin, Wolfram right to reply.

Date: Thu, 29 May 2003 13:55:02 -0400

From: Ed Fredkin

To: Stephen Wolfram

Stephen,

I carefully and clearly told the author of the NS article that to my
knowledge it is not true "... that Wolfram first became familiar with

cellular automata at Fredkin's lab in the '70's..." and further that you already knew about CA's.

My guess is that magazines see value in controversy and they would like to attribute statements to each of us that helps them titillate their readers. I tried in every way I could to correct any wrong impressions the author had. But what they end up doing is beyond my control.

As to cracking the fundamental theory of physics, I did read and did understand what you wrote about in NKS, however my interests lie in models that are regular and based on a simple underlying Cartesian lattice. The models I have been working on for the past few years are called "Salt" as they are CA's similar to an NaCl crystal. You can read about it at www.digitalphilosophy.org.

My approach to being consistent with QM, SR and GR is related to the fact that CA models of physics can exactly conserve such quantities as momentum, energy, charge etc. By means of a variant of Noether's Theorem, the physics of such CA's can exhibit all the symmetries we currently attribute to physics, but doing so asymptotically at scales above the lattice.

Thus, in my concept of a theory of physics, translation symmetry, rotation symmetry etc. would all be violated as we currently understand is true for time symmetry, parity symmetry and charge symmetry.

No one suggests that you should agree with all my ideas, however your comment in your prior email to me is unnecessarily condescending:

*> "I think it's a great pity that you've never taken
> the time to learn the technical details of physics as it's
> currently practiced. There's a lot known. And if you understood
> it, I think you'd be able to tell quite quickly which of
> your ideas are totally naive, and which might actually be interesting."*

What is certain is that there's no "great pity" necessary. I actually do know a lot about the technical details of physics. In any case, thirty years ago Feynman thought that I needed to learn more about certain aspects of QM. He was specific in what he felt was everything more that I needed to know (in order to make progress with my CA ideas). He offered to work with me, which was accomplished during the course of the year I spent at Caltech (1974-1975). I studied, learned more about QM and passed the final exam that Feynman gave me.

While

we argued a lot, Feynman never accused me of having naive ideas.

As to NKS 2003, it doesn't make a lot of sense for me to come to be a member of the audience. If you would like me to participate in some meaningful way, let me know.

Best regards,

Ed F

And after that exchange, Ed and I basically went back to being as we had been before—having pleasant interactions, without any particular scientific engagement. And in a sense for many years I kept out of Ed's scientific way—not seriously working on physics again until 2019.

Since 2002 I'd been living in the Boston area, so Ed and I ran into each other more often. And although Ed's behavior over *A New Kind of Science* had disappointed and upset me, it gave me a better understanding of Ed as a human being, and a vulnerable one at that.

The Later Ed

It was always a little hard to tell just what was going on with Ed. In July 2003, for example, he wrote to me:

Subject: Gunkel

Date: Thu, 24 Jul 2003 19:07:56 -0400

From: Ed Fredkin

To: Stephen Wolfram

Stephen,

First I must apologize for this long letter. Pat Gunkel sent me an email telling of your visit. It prompted me (who hardly ever writes anything) to type up my thoughts for whatever they're worth.

...

You might be surprised at the number of wise and intelligent people who really appreciate Pat and his works. Yet after more than 30 years of

fitful, diverse yet nearly continuous support, Pat has come to a situation that, to him, looks like the end of the line.

...

I, unfortunately, am no longer in a position to personally provide the kind of modest support that Pat needs to continue his church-mouse kind of existence.

...

There is no doubt that Pat can be a difficult person to help, but I notice that he has mellowed with age. Of course, Wolfram Research is not a charitable institution. But I believe that Pat's ideas on ideonomy are really important and that those ideas may form the basis of interesting future applications. The point of all this is that if what Pat is doing seems interesting to you, some arrangement with Wolfram Research might make sense.

(True to form, Gunkel followed up with a very forthright note, including a scathing critique he'd written of *A New Kind of Science*—as well as of Ed's theories. That wouldn't have deterred me, but I couldn't see anything Gunkel could actually do for us, so I never pursued this.)

But did Ed's note imply that Ed was running out of money? I'd always assumed some kind of vast business empire lurking in the background, but now I wasn't sure.

I saw Ed only a few times in the next couple of years—at events like a Festschrift for Sulak and a bat mitzvah for one of Feynman's granddaughters. But as usual, he was eager to tell stories, some of which I hadn't heard before—mostly about things far in the past. He said that in the early 1960s John Cocke had stolen the idea of RISC architecture from his murdered friend Ben Gurley, though it had taken him two decades to get it taken seriously. He said that around the same time he'd been pulled in by the Air Force to help with analysis of blast waves from nuclear tests (and that story came with descriptions of B-52s doing loop-the-loop maneuvers when they dropped atomic bombs). He said that he'd once demoed the Muse music system (which, he emphasized, he, not Minsky, had invented) to an astonished audience in the Soviet Union. He said that he'd advised Richard Branson on his transatlantic balloon trip, telling him his butane burners weren't correctly mounted—and in fact they fell off. And so on.

In 2005 Ed told me he'd been working with a programmer in California named Dan Miller (who'd developed audio compression software [and been at the NKS 2003 conference that Ed had been so upset about]) on the new 3D cellular automaton he'd invented that he called the "SALT architecture" because its pattern of updates were like the Na and Cl in a salt crystal.

But then in 2008 Ed told me he'd sold his island—presumably relieving whatever financial issues he'd had before—and suddenly Ed started to show up much more. He told me (as he did quite a few times) that he was working on a book (which never materialized). He told me he was teaching a course at Carnegie Mellon on the "Physics of Theoretical Computation"—which was apparently actually a very-much-as-before "engineering-style" effort to explore building features of physics from a cellular automaton, now with his SALT architecture. He invited me to a dinner at his house in honor of 't Hooft, photographed here with Ed, me and Sulak:



That fall, Ed came to the Midwest NKS Conference in Indiana, here photographed in a discussion with Greg Chaitin, me and others (Toffoli was supposed to be there too, but didn't show up):



I would interact with Ed quite regularly after that—most often with him telling me about his use of Mathematica and soon [Wolfram|Alpha](#). In 2012 Ed—now aged 78—sent me a nice “I have an idea” email (I made the requested introduction, though I’m not sure if this ever went anywhere):

Subject: Alpha and Problem Solving
Date: Fri, 19 Oct 2012 20:12:01 +0000
From: Edward Fredkin
To: Steve Wolfram

Steve,

The first thing I taught at MIT was a course in general problem solving (in 1968).

I’m now developing a new course on General Problem Solving which I expect to offer first at Harvard’s HILR program. Part of the motivation came from watching Joyce struggle with a Harvard course on Chemistry, where a lot of the homework involved units conversions. I noticed that Alpha promptly solved many of Joyce’s homework problems including some involving chemical reactions. (The course was really for students planning to take the MCAT Exam in order to get into Medical School). One clue that you might give to the Alpha developers, is to work toward getting Alpha to have more of the capabilities necessary to pass different standard tests that involve various kinds of quantitative analysis. (Of course, you

might
have already done so.)

You might recall that I discussed the issue of units conversion with you long ago (before Mathematica), and you described the idea you then had that turned into Convert in Mathematica.

In any case, Alpha is fantastic, and getting better all the time. My plan is that every one of my students must use Alpha for every problem that involves numbers, along with some that don't involve numbers. My motto is John McCarthy's dictum: "Those who refuse to do arithmetic are doomed to talk nonsense!" However, with Alpha, the problem solver doesn't have to do the arithmetic or the units conversions; Alpha can do it!

It would be helpful if I could get a little bit of cooperation from someone in the Alpha group. Basically, I will want to talk to an Alpha expert from time to time to make sure I'm taking advantage of the best that Alpha can do along with resources already developed for introducing Alpha to new users. My initial students will be drawn from a group of retirees who, while clearly above average in intelligence, may have few recently used skills in mathematics. I also expect that almost all of my initial students will be first time Alpha users. Again, I might profit from discussion with someone who has thought about how to introduce Alpha to beginners.

Let me know what you think or, if you like, we could get together to talk about it.

Best regards and Congratulations!

Ed

In 2014, when I recorded some oral history with Ed—now age 80—he was again brimming with ideas. The one he was most excited about had to do with weather prediction. It started from the observation that most smartphones have pressure sensors in them. Ed’s idea was to use these—and more—to create a sensor net that would continuously collect billions of pressure measurements, to be fed as input to weather forecast codes. Channeling his lifelong interest in reversible computing he imagined that the codes could be made reversible, and that running backwards from an incorrect prediction could tell one where more data had to be collected. Then Ed imagined doing this by having tiny balloons all over the place—with nothing that would cause trouble if a plane ran into it. He had a whole plan for partners he wanted to get (and, yes, he wanted us to be part of it too). And in typical Ed fashion, it was all laced with stories:

You know, I had this personal experience with weather. I was flying a glider along at 16,000 feet, and I encountered sink. You know, sink is wind blowing down. And the speed of the sink was 10,000 feet a minute. I was at 16,000 feet. And two minutes later, I was on the ground landing. Not on purpose. You know my attitude was—if I don’t see a big grading on the ground—[the wind] can’t keep going this way all the way down, so I won’t be killed. Actually, in that same storm, one of the pilots was killed.

...

The weather people just aren’t into the vertical movement of air. They do everything in layers. But this went through a lot of layers all at once in an organized fashion. So the point is that to talk about thousands or even millions of sensors makes no sense. You’re not going to do good weather until you get billions of sensors. That’s my opinion.

We talked about whether sensitive dependence on initial conditions destroys all predictability in fluid dynamics. I have theoretical and computational reasons to think it doesn’t. But Ed had a story:

There’s a mountain in California I happen to know, and I have a picture of a cloud street that starts on that mountain because it has a very peculiar geometry, and then runs for 2,000 miles.

So this particular mountain has an area of its rock that faces towards the east and it’s big. And what happens is when the Sun is shining on that and the wet wind is coming from the Pacific and so on, you get this big cumulus cloud that flows back this way, and then you get another one and it pulses. You get one after another. And these are very stable things and they travel a very long way. So my point is that amidst all the randomness there’s a lot of order that can be found and understood. There are regions that have funny properties. They’re much more temperature stable. There’s like islands of stability. And things like that get ignored by everything people are doing today, you know what I mean?

I would send things I'd written to Ed. I didn't really think he'd read them. But I thought he might at least enjoy their concepts. And often he would respond with ideas of his own. I sent him an announcement about our [Tweet-a-Program](#) project (now reconfigured because of Twitter changes) with the one-line comment (reflecting his "best programmer" self-characterization): "A new frontier of programming prowess?" He responded, in typical Ed fashion, with an idea—that's actually a little reminiscent of modern AI image generation:

Subject: Re: Tweet-a-Program
Date: Fri, 19 Sep 2014 21:25:47 +0000
From: Edward Fredkin
To: Stephen Wolfram

Hi,

I like it! As usual, it gave me ideas that might be outside of your current concept.

We should talk sometime, so that I can explain something closely related to [Tweet-a-Program] but decidedly different and perhaps even more fun. Strangely, it has to do with Haiku.

What I have figured out is that there could be a new kind of Haiku, where the text is interpreted by Mathematica to generate an image.

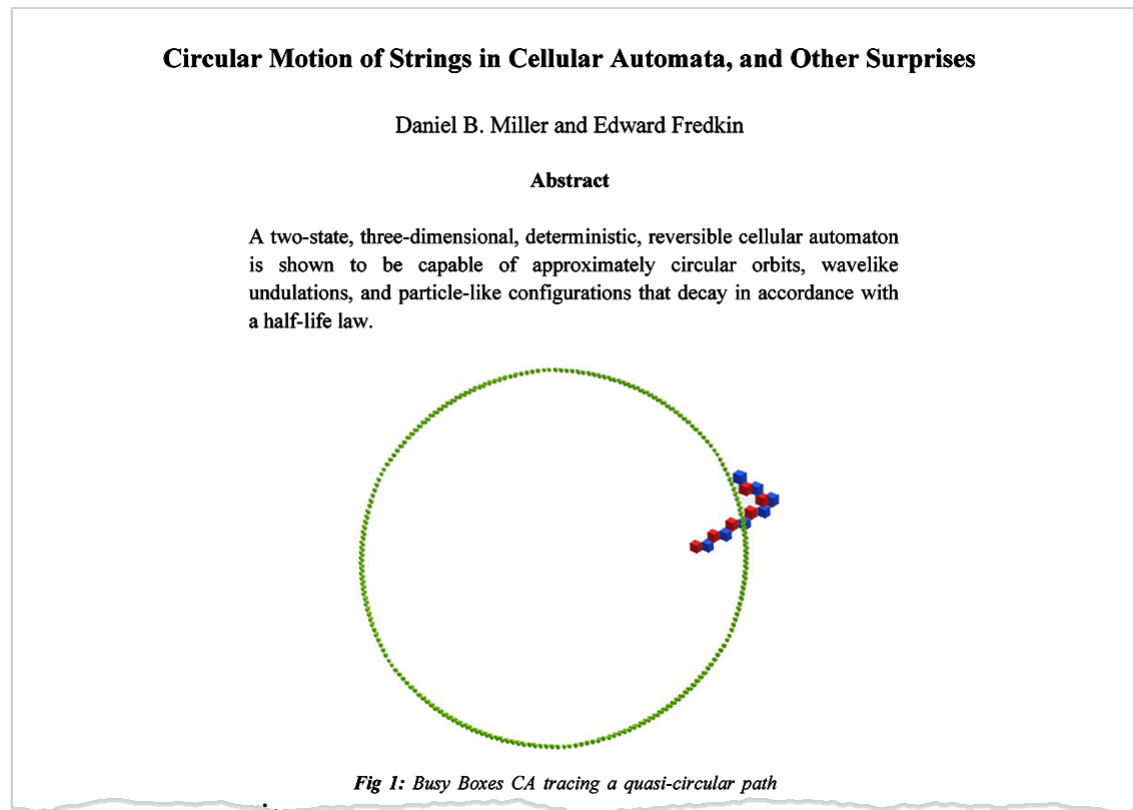
...

The trick will be having the image reflect something of the Haiku meaning, even if only abstractly. I don't know how to do this so that it does the perfect thing every time, but I have thought of something that could be fun, and a person could become skilled at creating Mathematica Haikus that seem to reflect some aspects of the feeling of the words in an image with some increasing probability of doing it well, as a result of practice.

...

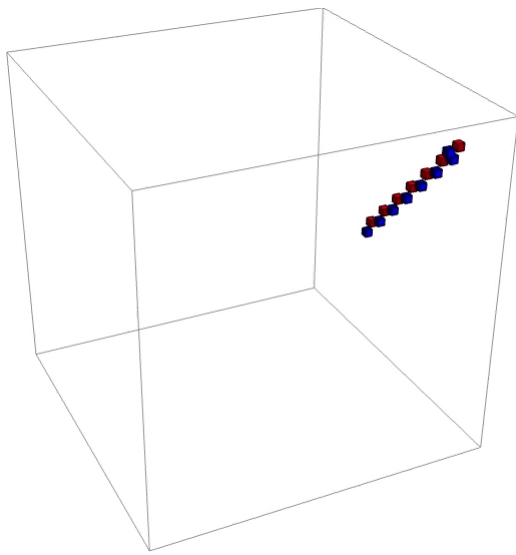
Ed

Late in 2014 Ed sent me another piece of mail saying he was starting a project to produce a “new cellular automaton system”—and he wanted to use our technology to do it. He also sent me a [paper he’d written](#) about his SALT cellular automaton:

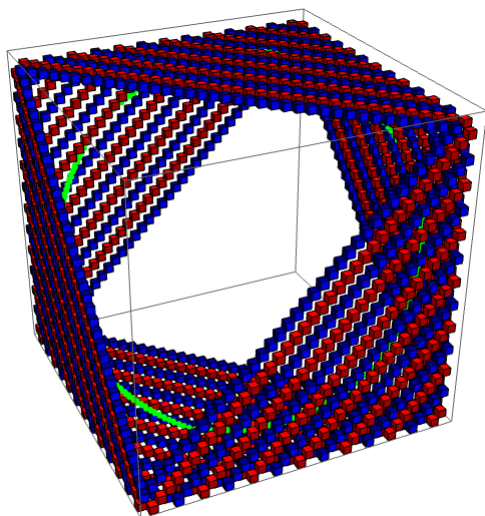


Finally—and without my help—Ed seemed to have mastered the art of academic papers. This one was on the arXiv preprint server. Others—with titles like “[An Introduction to Digital Philosophy](#)”—had appeared in academic journals. (Ones with titles like “[A New Cosmogony](#)” and “[Finite Nature](#)” were more privately circulated.) But what most struck me about this particular paper was that—for the first time—it seemed to have actual images of cellular automaton behavior. Ever since those few minutes with the PERQ computer on Ed’s island in 1982 I hadn’t seen Ed ever show anything like that. And now Ed was again chasing that old question Minsky had asked, of making a circle with a cellular automaton.

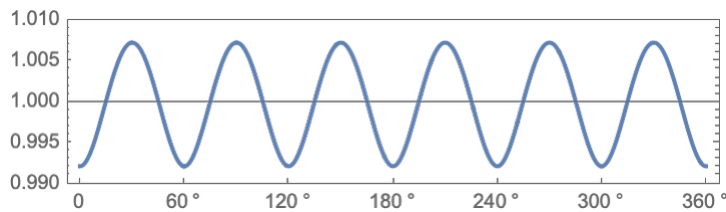
At the time, I didn’t have a chance to see what Ed had actually done, and whether he’d finally solved it. But in writing this piece, I decided I’d better try to find out. The actual rule—that Ed and Dan Miller called “[BusyBoxes](#)”—is quite complicated, involving knight’s-move neighborhoods, etc. Their claim was that starting with a string of cells in a particular configuration, the average of their positions would trace out what in the limit of a long string would be a circle:



At first it looks like a kind of magic trick (and no, nothing is bouncing off any “walls”; the direction changes are just a consequence of the initial pattern of cells). But if you keep all the locations that get visited, things start to seem less mysterious—because what you realize is that the “basket” that gets “woven” is actually just a cube, viewed from a corner:



Where does the apparent circle come from? The details are a bit complicated—and I’ve put them in an appendix below. But suffice it say to that Ed’s old nemesis—calculus—comes in very handy. And in fact it lets one show that although one gets almost a circle, it’s not quite a circle; even with an infinite string, its radius is still wiggling by about 0.5% as one goes around the “circle”:



And—as we’ll see below—remarkably enough one can get a closed-form result for the amount of wiggleness (here computed as the ratio of maximum to minimum radius):

$$\sqrt{3} - \frac{\sqrt{7}}{3} + \frac{1}{4} \log\left(\frac{7}{3} + \frac{4}{\sqrt{3}}\right) + \frac{1}{2} \log(\sqrt{7} - 2) \approx 1.01529$$

In earlier years, Ed might have tried to say that generating a circle (which this doesn’t) was tantamount to showing that a cellular automaton could reproduce physics. But by now I think he realized that it was really much more complicated than that. And he wasn’t mentioning physics much to me anymore. But—perhaps not least because many of his longtime interlocutors had by then died—he was interacting with me more than before. And perhaps he was even beginning to think that I might have a bit more to contribute than he’d assumed.

In December 2015 I sent Ed a [piece I’d written](#) to celebrate the bicentenary of Ada Lovelace, and he responded:

Date: Fri, 11 Dec 2015 15:58:14 +0000

From: Edward Fredkin

To: Stephen Wolfram

Stephen,

I was truly blown away by your essay re Ada Lovelace! You’ve got a lot more to give the world than I had imagined, and I, more than anyone else, appreciate what you might still be capable of accomplishing.

It’s too bad that some persons at MIT, for far too long, hung onto one dimensional views focussed on what Macsyma might have been. My own impressions have always been different, I recognized your potential long ago and consequently invited you to one of my Mosquito Island conferences some 3.5 decades ago.

In any case, much of what Mathematica makes possible is very important and valuable to me. As you know I was an early user and continue to be a user.

Many of my interests have run along many paths opened up by activities you have instigated at Wolfram. Wolfram Alpha and its connections to Siri, are examples.

Your new book “An Elementary Introduction to the Wolfram Language” (I don’t yet have a hard copy) fits in with a project I had in mind for my grandchild Robert, who at age 6 already seems to be extraordinarily talented mathematically.

To cut to the chase, I want to make a proposal: Although I’m too old to be a regular employee, I’d nevertheless like to have an association with Wolfram, where I might be able to contribute ideas, and solve problems (I’m still quite good at that).

I won’t need much from you other than your opening the door to my involvement at Wolfram. What I have in mind would be an arrangement where I could work for Wolfram, with some kind of arrangement other than full time employment.

I’ve attached something I wrote recently.

Ed

Gosh! That was an unexpected development. Flattering, I suppose. But my main reaction was a kind of sadness. Yes, after all these years, Ed had finally read something I’d written. But somehow his response sounded like he was surrendering. This wasn’t the “I-want-to-do-everything-for-myself” Ed I had known all this time. This was an Ed who somehow felt he needed us to support him. And while our company has been able to absorb a great many “unusual” people—with terrific success—Ed seemed like he was pretty far outside our envelope.

At the time, I didn’t look at the attachment Ed sent with his email. But opening it now adds to my sense of sadness. It was a 13-page document about a system Ed imagined that would

help people with “various forms of cognitive disabilities”, including a section on “Dementia and Alzheimer’s”:

Assist

Assist is a prosthetic system that helps persons, even those with various forms of cognitive disabilities, to function at or above their normal unassisted capabilities. A most important feature of *Assist* is its constant ability to precisely evaluate and model the current mental status, of an individual, whom we will refer to as “*User*”. There are ups and downs in every *User*’s requirements for assistance, and *Assist* can be very sensitive, accurate and responsive to the current state, capabilities and needs of *User*. *Assist* is a new kind of prosthetic system that can provide real time, invaluable assistance to normal persons, but especially to persons with various degrees of temporary or permanent mental or sensory impairment. While potentially useful for almost every normal person, it is specifically designed to help those who suffer a temporary (or permanent), partial (or total) loss of various capabilities in one or more areas of sensory perception, memory or intellectual cognition.

There are standardized tests to measure various cognitive capabilities. Cognitively impaired subjects usually get scores that reflect their handicaps. What matters is that most of those same persons, with *Assist*, could score above the current average of normal individuals on those same standardized tests. Most importantly, they could also be enabled, by *Assist*, to experience practical and intellectual life, in general, at the high end of what we currently call “Normal”.

Dementia and Alzheimer’s

Assist will have a special signal it gives to *User* to suggest the quick termination (and recovery) of what seems to be *User* starting the kind of repetition episode characteristic of dementia and Alzheimer’s. This common behavior occurs when *User* has something to say, then says it, and a short while later user remembers again what they wanted to say while not remembering that they already said it. In such cases, *User* can react in one of two ways: instant acceptance of *Assist*’s observation and signal, by hesitating and not continuing with that line of discourse, the other is a kind of challenge to *Assist*, to take a moment to privately whisper to *User* sufficient text or audio to convince *User* of the validity of *Assist*’s observation.

Assist keeps an informational model of everything that has been communicated from *User* to each person *User* knows. Thus, when *User* starts a line of conversation that *Assist* can determine will be an attempt to again communicate something already recently communicated to that same person, *Assist* warns *User* in a subtle way that allows *User* to avoid that action or comment. This

It wasn’t until 2017 that Ed explicitly mentioned to me that his short-term memory was failing—though in talking to him it had been increasingly obvious for several years. He said he’d joined a group of people who were writing their memoirs. I told him I’d look forward to seeing his, though I’m not sure he ever made much progress on them.

Ed continued to send me ideas and proposals. There was a very Ed-like “global idea” about creating a system “GM” (presumably for “General Mathematician”) that would effectively “learn all of mathematics” by automatically reading math books, etc. (yes, definite overtones of what’s happening with LLM-meets-Wolfram-Language):

Letter to Stephen.

I woke up this morning with a few random thoughts and ideas... some of which might prove useful. While it's obvious that much of what's here must have occurred to you and others at Mathematica... it's also likely that something might be useful.

I believe that it's time to tackle the biggest problem in Mathematics!

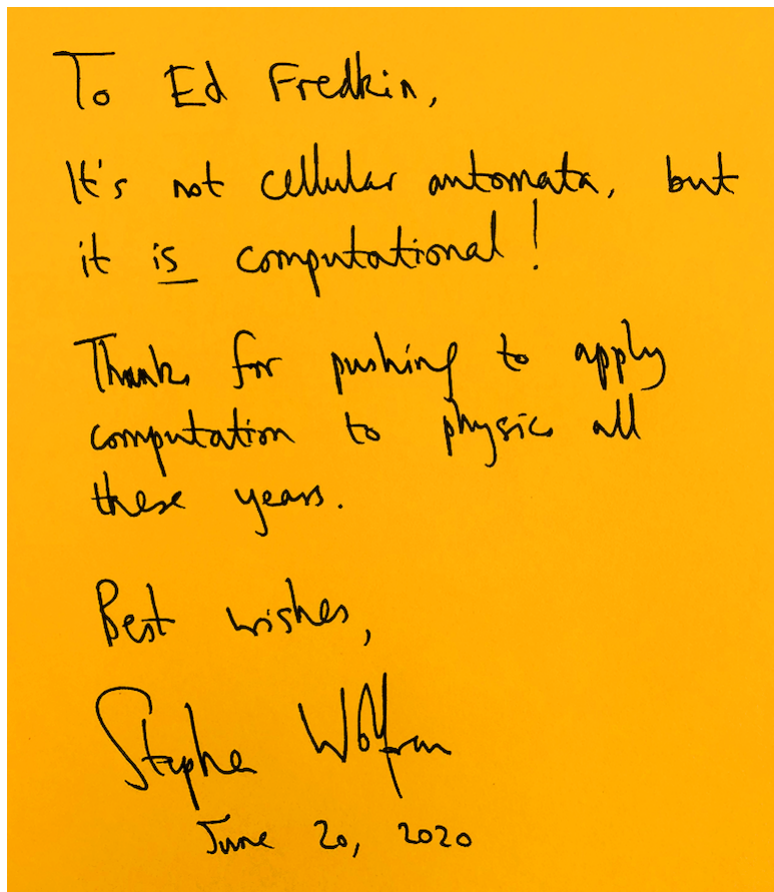
It should be possible to apply given results from work on Mathematica, Big Data technologies, and a few new ideas so as to create a system, "GM", that could do much of the following:

- I. Automatically read proofs while understanding them at a high level and being able to incorporate them into the kind of more global knowledge that could be useful to GM.
- II. Automatically read Mathematics Books... Euclid, Courant and everything else appropriate.
- III. Essentially *Understand* everything that is read (insofar as "Learning" the mathematics and "Understanding" the proofs... and, of course, checking them) and finally incorporating new knowledge into the body of capabilities that can be applied to both

GM could have the capabilities necessary to interactively explain either a proof to some individual, or teach an individual person some aspect of mathematics. This would require GM to have or create a model of that person, with regards to what that person knows and doesn't know, and various measures that relate to that individual's mathematical capabilities. GM should maintain a library of all mathematicians with details as to what they know and the kinds of problems they are best at. GM should be able to interact with various mathematicians, when it needs help, and conversely make itself available to those same mathematicians when they need help. In essence, GM could and should become every Mathematician's best and favorite colleague!

Later there were several pieces of mail about a new idea for factoring integers. In the first of them (from 2016), Ed told me that when the NeXT computer first came out (in 1989) he'd used Mathematica on it to simulate a reversible hardware multiplier. And being reminded of this by a historical piece I'd written, he said it had "started me thinking, again, about that problem and I had a new insight that appears to so greatly reduce the complexity of a reversible multiplier so as to possibly make it better at factoring large integers than current algorithms." He wrote me about this several more times, suggesting various kinds of collaborations. Finally, in 2018 he told me how the method worked, saying it involved doing reversible arithmetic using balanced ternary. (Strangely enough, years earlier Ed had told me about Soviet computers that also used balanced ternary.)

I think that was the last technical conversation I had with Ed. A couple of years later I sent him the book about our Physics Project with the inscription:



And I would see him at least once a year at the Boston-area physics get-together organized by Boston University. He would always tell me stories. Often the same stories, and sometimes stories about me. And indeed as I was writing this piece I actually found a [video Ed made in November 2020](#) that has such a story, albeit by this point seriously muddled (and, no, I've basically never "run" a cellular automaton by hand in my life!):

I used to organize meetings in the Caribbean and I did this because I had an island in the Caribbean ... I invited Wolfram to come down. Wolfram had done pioneering work in cellular automata. ... He was a great guy, you know, and I wanted him to get on the bandwagon ... He shows up at the meeting and he had done all his work by hand as had everyone else in cellular automata. He didn't think of using a computer. [!] I had a display processor that I modified to be able to run a cellular automaton with the stuff that it used to put text up on the screen. And so I'm showing him a cellular automata running at 60 frames a second continuously like a movie. This was 10,000 times faster than doing it by hand which is what he'd always done. He never thought of using a computer to do cellular automata and he turns around and walks out and and he left the island and went back to someplace else. So [later] I went to his meeting at Los Alamos and I ran into him again and he was now doing computer work. And I said to him "How come in all your work you don't have a reversible [rule]", and he says to me "Oh, reversible ones are all trivial". And I went up and this is the most telling thing about his intellect: he's a very smart guy [and when I] showed him how he could change his rule slightly and make it reversible his eyes just about popped out of his head and he knew I was correct.

I may have introduced him to this field but what he has done is he is far better than I at getting other people involved. I've never bothered and I don't have the talent that he has for that. What he did was he came up with similar ideas and initially he didn't give me the credit I thought I deserved. But it became apparent to me that he did this independently and he's better at writing things and better at hiring bright people who can do things than I ever was.

And right after that, Ed ends the video with:

As I look back on my career I've had a fantastic life and I'm not unhappy about any aspect of it because, you know, I've accomplished everything I might have done and in spite of various handicaps—like not being a writer—I still have done a lot and the world, uh, understands me, I think, and appreciates what I've done.

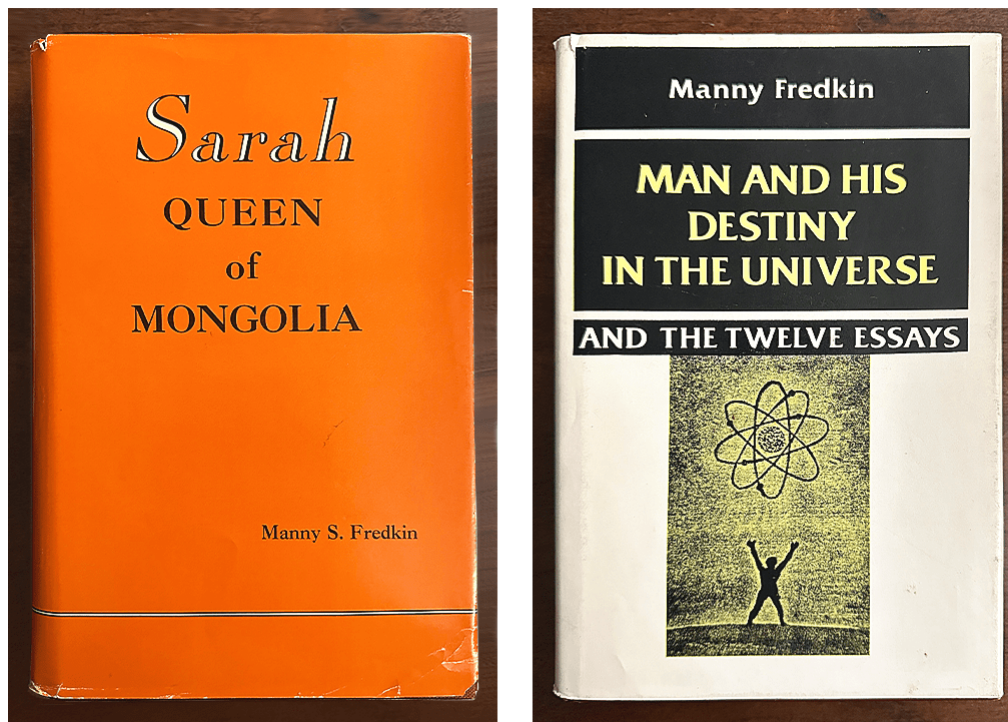
When I saw Ed in 2022 he wasn't able to say much. But, though it was a struggle, he was keen to make one point to me, that seemed to matter a lot to him: "You've managed to get people to follow you", he said "I was never able to do that". I saw Ed one last time this May. Joyce explained that Ed had "bumped his head", and, in a very Ed-like way, she was avoiding a repeat by getting him to wear a bike helmet. She wanted someone to snap a picture of me and her with Ed:



Six weeks later, Ed died, at the age of 88.

I went to see Joyce and Rick a few weeks later, among other things to check facts for this piece. I'd heard from Ed that his ancestors had provided wood for the imperial palace in St. Petersburg. But I'd also heard from someone else that Ed had said he was descended from Mongolian royalty. And as I was about to leave, I thought I might as well ask. "Oh yes", they

said. “And Ed’s father even wrote a historical novel about it”. And they showed me two books (both from the mid-1980s):



I’m not sure who Sarah, Queen of Mongolia was, but the book blurb claims that Ed’s father was her great-great-great-grandson—and goes on to speak of the “strong family inheritance of a mind that analyzes not only the injustice of human oppression but offers realistic and beneficial solutions”.

Summing Up Ed

“Can that really be true?” I often asked myself when hearing yet another of Ed’s implausible stories. And of course it didn’t help that stories he told—even to me—about me weren’t true. But the remarkable thing in writing this piece is that I’ve been able to verify that a lot of Ed’s stories—implausible though they may have sounded—were in fact true. Yes, they were often embellished, and parts that didn’t reflect so well on Ed were omitted. But together they defined a remarkable tapestry of a life.

It was in many ways a very independent life. Ed had friends and family members to whom he stayed close throughout his life. But mostly it was “Ed for himself, against the world”. He didn’t want to learn anything from anyone else; he wanted to figure out everything for himself. He wanted to invent his own ideas; he wasn’t too interested in other people’s. In a rather Air-Force-pilot kind of way (“eject or not?”) he liked to be decisive—and he liked to

be incisive too, always figuring out a clear, simple thing to say. Sometimes that came across as naive. And sometimes it was in fact naive. But mostly Ed didn't seem to mind much; he would just go on to another idea.

Ed was a great storyteller, and an engaging speaker. For some reason he developed the theory that he couldn't write—but there's ample evidence, going back even to his teenage years, that this wasn't true. If there was a problem, it was with content, not writing. And the issue with the content was that it tended to just be too Ed-specific—too insular—and not connected enough for other people to be able to understand or appreciate it.

I don't know what Ed was like as a manager; I rather suspect he may have suffered from trying to be a bit too clever, with too many ideas and too much gamification. In the end, he felt he'd failed as a leader, and perhaps that was inevitable given how independent he always wanted to be. Despite his stints as an academic administrator and as a CEO, Ed was in the end fundamentally a lone warrior (and problem solver), not a general.

And what about all those ideas? Most never developed very far. Some were pretty wild. But many had at least a kernel of visionary insight. The details of the universe as a cellular automaton didn't make sense. But the idea that the universe is somehow computational is surely correct. And spread over the course of more than six decades, Ed spun out nuggets of ideas that would later appear—usually much more developed—in a remarkable range of areas.

Ed projected a kind of personal serenity—yet he was in many ways deeply competitive. Most of the time, though, he was able to define the arena of his competitiveness so idiosyncratically that there really weren't other contenders. And I think in the end Ed felt pretty good about all the things he'd managed to do in his life. It was fitting that he owned an actual island. Because somehow an island was a metaphor for Ed's life: separate, independent and unique.

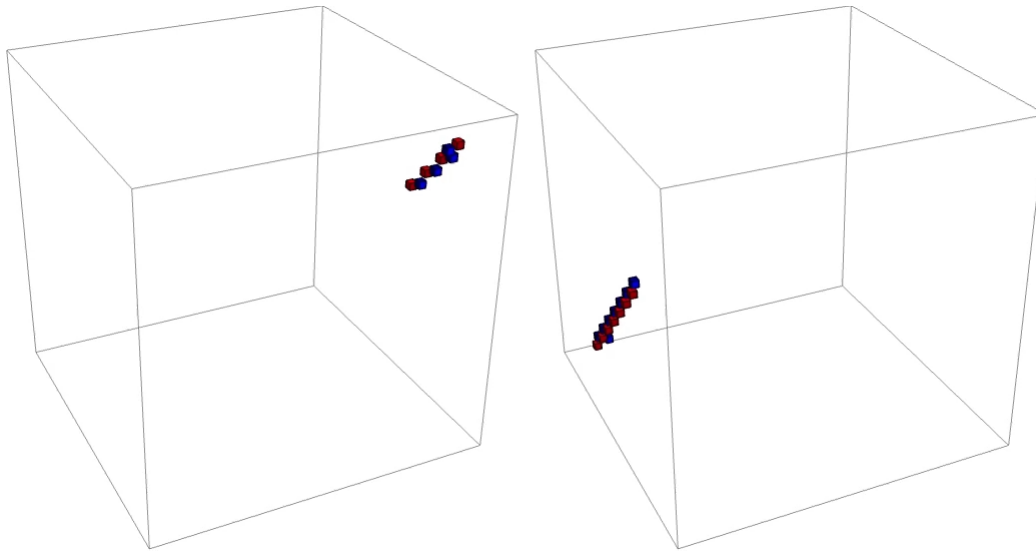
Thanks

I've had help with information for this piece from many people, including Joyce Fredkin, Rick Fredkin, Simson Garfinkel, Andrea Gerlach, Bill Gosper, Howard Gutowitz, Steven Levy, Norm Margolus, Margaret Minsky, Dave Moon, John Moussouris, Mark Nahabedian, Walter Parkes, David Reiss, Brian Silverman, George Sulak, Larry Sulak and Matthew Szudzik. (Tom Toffoli agreed to talk, but didn't show

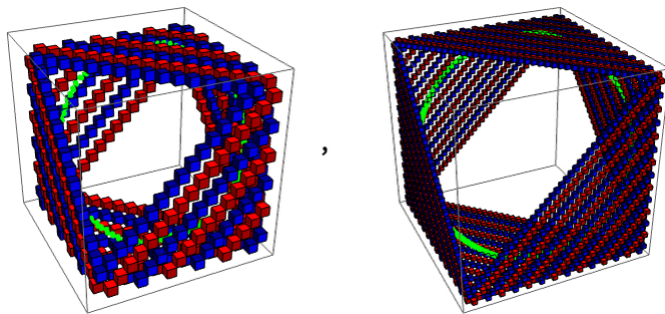
up.) I thank the Department of Distinctive Collections at the MIT Library for access to the Fredkin papers archive there. Thanks also to Brad Klee and Nik Murzin for technical help.

Appendix: Analyzing the Not-Quite-Circle

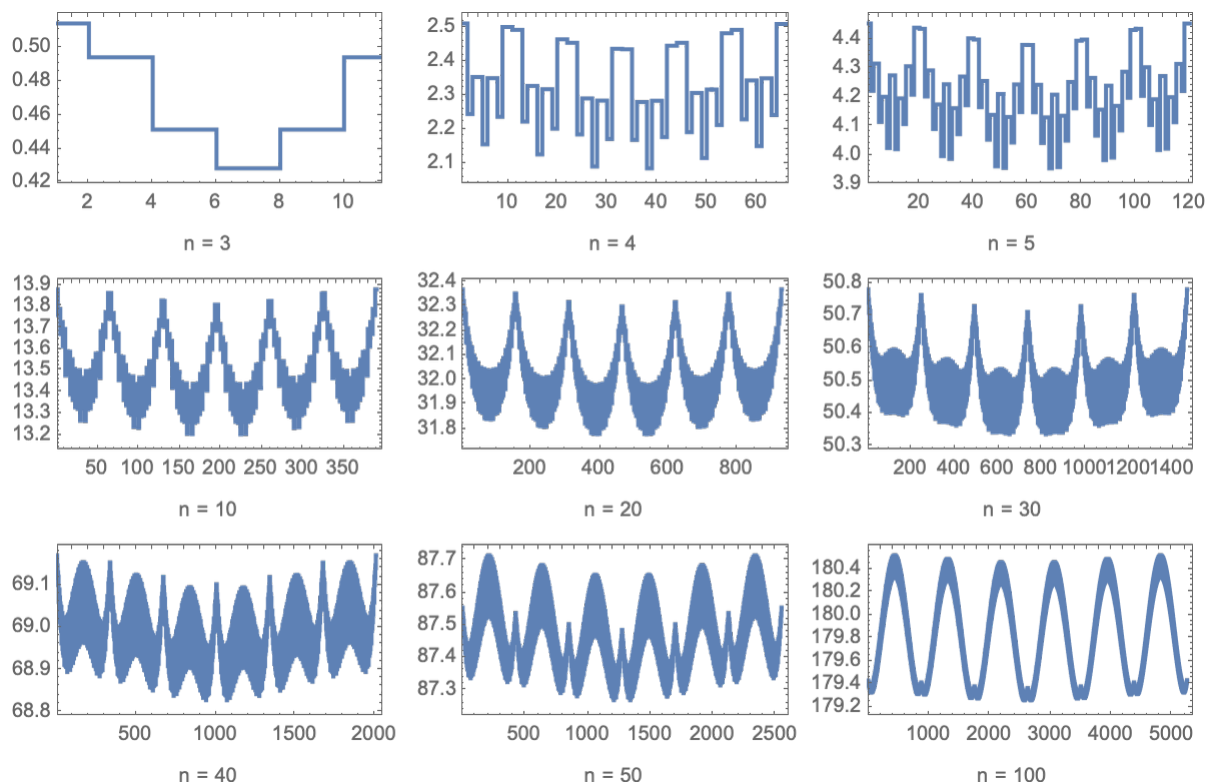
Here’s what the **SALT cellular automaton** does for two sizes of initial “string”:



For an initial string of length n (with $n > 2$), the overall period is $54n - 43$, and the envelope “woven” going through all configurations is:



The “circle” is obtained by averaging the positions of all cells present at a given time step. The “circle” is always planar, but its effective radius varies with direction (i.e. as the system steps through each cycle):



Ed and Dan Miller looked at the standard deviation of the effective radius as a function of n , computing it up to $n = 20$, and getting the following results:

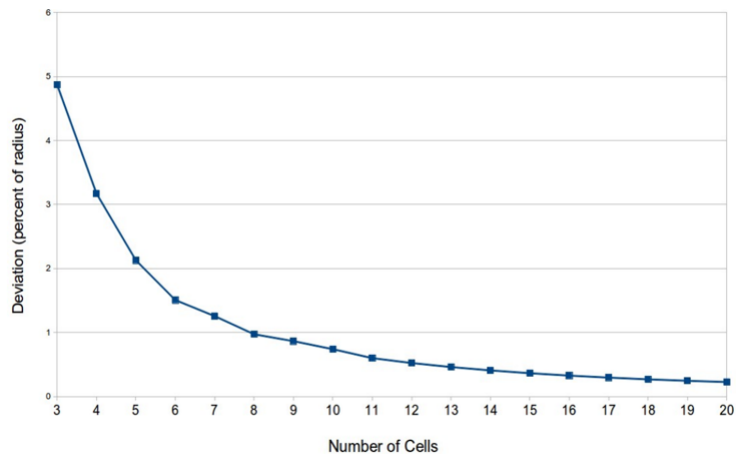
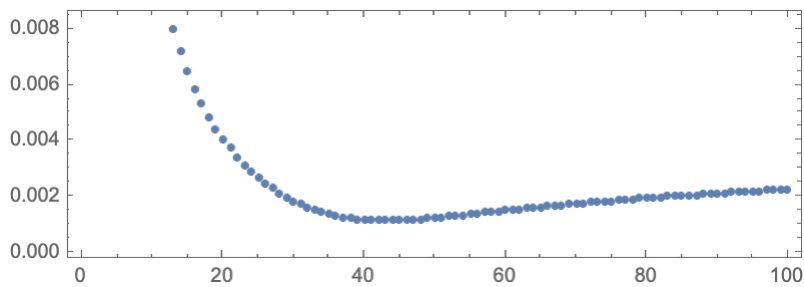


Fig 10: Circularity increases with cell count.

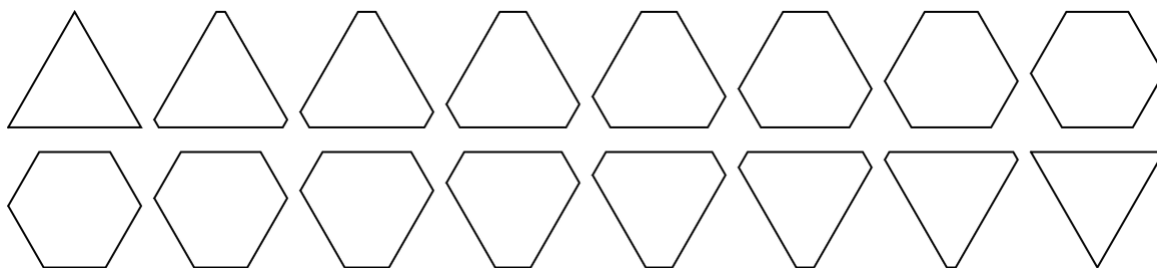
It looked as if the standard deviation was just going to go smoothly to zero—so that for an infinite string one would get a perfect circle. But that turns out not to be true, as one can see by extending the computation to slightly larger values of n :



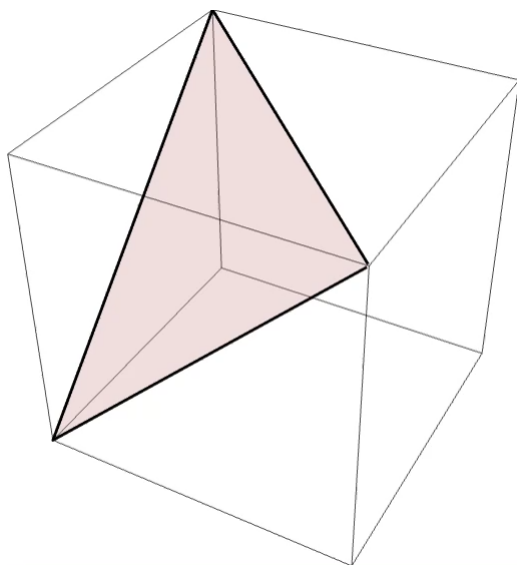
And actually there's a minimum at $n = 43$, with standard deviation 0.0012 (and fractional size discrepancy 0.0048)—and it doesn't look like even for $n \rightarrow \infty$ one will get a perfect circle.

But how can one work out the $n \rightarrow \infty$ case? It's actually a nice application for calculus.

First, notice that the “basket” consists of a series of layers of a cube viewed from one of its corners, or in other words a sequence of shapes like this:



Here's how these are formed as one sweeps through the cube:



One can think of the string in the cellular automaton as spanning these “layers”, and successively moving around all of them as the cellular automaton evolves. In the continuum limit, there's effectively a parameter t that defines where on each “layer curve” one is at a

particular time. Conveniently enough, the length of all the layer curves is the same (for a unit cube it is $3\sqrt{2} \approx 4.2$). With successive layers parametrized by a variable s (running from 0 to 1) the corners of the layer curves (all normalized to have length 1) are given by:

$$\left\{ \left\{ 0, \frac{s}{3\sqrt{3}} \right\}, \left\{ \frac{s}{6}, -\frac{s}{6\sqrt{3}} \right\}, \left\{ \frac{2-s}{6}, -\frac{s}{6\sqrt{3}} \right\}, \left\{ \frac{1}{3}, \frac{s}{3\sqrt{3}} \right\}, \left\{ \frac{s+1}{6}, -\frac{s-3}{6\sqrt{3}} \right\}, \left\{ \frac{1-s}{6}, -\frac{s-3}{6\sqrt{3}} \right\}, \left\{ 0, \frac{s}{3\sqrt{3}} \right\} \right\}$$

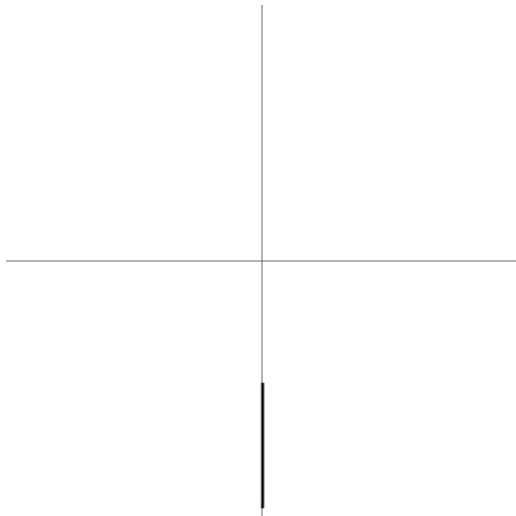
Now we need to find the actual x, y positions of string elements (AKA infinitesimal cells) as a function of s and t . Since the edges of the layer polygons are always straight, in each of a series of “piecewise regions” in s and t (with breakpoints defined by the corners of the polygons), we get expressions for x and y that are linear in s and t :

$$\left\{ \begin{array}{ll} \left\{ \frac{s}{6} + t, -\frac{s}{6\sqrt{3}} \right\} & s + 3t \leq 1 \\ \left\{ \frac{1}{6}(3t + 1), \frac{2s+9t-3}{6\sqrt{3}} \right\} & (s = 1 \vee s + 3t \geq 1) \wedge 3t \leq 1 \\ \left\{ \frac{1-t}{2}, \frac{2s+9t-3}{6\sqrt{3}} \right\} & 3t \geq 1 \wedge s + 3t \leq 2 \\ \left\{ \frac{1}{6}(-s - 6t + 5), -\frac{s-3}{6\sqrt{3}} \right\} & s + 3t \geq 2 \wedge 3t \leq 2 \\ \left\{ \frac{1}{6}(-s - 3t + 3), -\frac{s+9t-9}{6\sqrt{3}} \right\} & 3t \geq 2 \wedge s + 3t \leq 3 \\ \left\{ \frac{1}{6}(s + 3t - 3), -\frac{s+9t-9}{6\sqrt{3}} \right\} & \text{True} \end{array} \right.$$

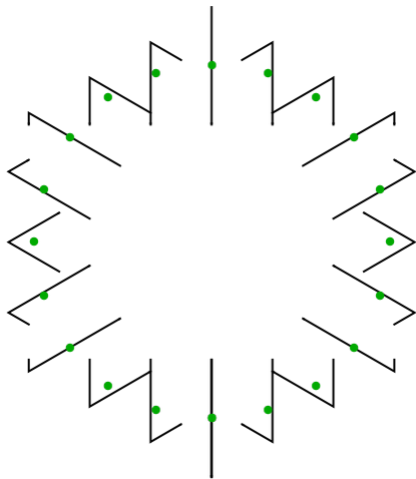
One subtlety is that the string in essence turns as time progresses, so that it effectively samples a different t value for different layers s . To correct for this, we have to find for which t we get $x = 0$ for a given s . It’s convenient to put the center of all our layer curves at $\{0, 0\}$, and we can do this now by subtracting $\left\{ \frac{1}{6}, \frac{1}{6\sqrt{3}} \right\}$. Then the (first) value of t at which $x = 0$ is given simply by:

$$t = \frac{1-s}{6}$$

The parametric surface we now get as a function of t is (with discrete lines indicating particular values of s):



Now we can slice the parametric surface not in discrete s values but instead in discrete t values—thus getting what’s basically a sequence of effective strings at discrete times:



The centroids of the strings are indicated in green, and these are then points on our potential circle. Using what we did above, the radius of this “circle” as a function of t can then be found by integrating $\sqrt{x^2 + y^2}$ over s . The result is algebraically complicated, but has a closed form:

$$\begin{aligned}
& \frac{1}{4\sqrt{3}} \\
& \frac{1}{12\sqrt{3}} \left(108t^2 \log \left(-\frac{(2\sqrt{9t(3t-5)+19}-1)(\sqrt{36t(3t-4)+49}-1)}{6(9(3-2t)t+\sqrt{18t(2t-3)+21}-11)} \right) + 6\sqrt{18t(2t-3)+21} - 2\sqrt{9t(3t-5)+19} - \right. \\
& \quad \left. \sqrt{36t(3t-4)+49} + (75-180t) \log \left(\frac{1-2\sqrt{9t(3t-5)+19}}{6t-2\sqrt{18t(2t-3)+21}-3} \right) + 48(3t-1) \log \left(\frac{6t+2\sqrt{18t(2t-3)+21}-6}{\sqrt{36t(3t-4)+49}-1} \right) \right) \\
& \frac{1}{12\sqrt{3}} \left(108t^2 \log \left(-\frac{(\sqrt{36t(3t-2)+13}-1)(2\sqrt{9t(3t-1)+1}-1)}{6(9(1-2t)t+\sqrt{18t(2t-1)+3}-2)} \right) + 6\sqrt{18t(2t-1)+3} - \sqrt{36t(3t-2)+13} - \right. \\
& \quad \left. 2\sqrt{9t(3t-1)+1} + (12-72t) \log \left(\frac{\sqrt{36t(3t-2)+13}-1}{2\sqrt{18t(2t-1)+3}-6t} \right) + (36t-3) \log \left(\frac{6t+2\sqrt{18t(2t-1)+3}-3}{2\sqrt{9t(3t-1)+1}-1} \right) \right) \\
& \frac{1}{12\sqrt{3}} \left(-\sqrt{108(t-2)t+109} - 2\sqrt{9t(3t-5)+19} + 6\sqrt{6t(6t-11)+31} - \right. \\
& \quad \left. 3(5-6t)^2 \log \left(\frac{6t+2\sqrt{6t(6t-11)+31}-7}{2\sqrt{9t(3t-5)+19}-1} \right) + 108(t-1)^2 \log \left(\frac{\sqrt{108(t-2)t+109}-1}{2(-3t+\sqrt{6t(6t-11)+31}+2)} \right) \right) \\
& \frac{1}{12\sqrt{3}} \left(-108t^2 \log(-6(3(7-6t)t+\sqrt{6t(6t-7)+13}-7)) - 2\sqrt{27(t-1)t+7} - \sqrt{36t(3t-4)+49} + \right. \\
& \quad 6\sqrt{6t(6t-7)+13} + 12(2-3t)^2 \log(\sqrt{36t(3t-4)+49}-1) + 27(1-2t)^2 \log(2\sqrt{27(t-1)t+7}-1) + \\
& \quad 36t(4\log(-3t+\sqrt{6t(6t-7)+13}+1) + 3\log(6t+2\sqrt{6t(6t-7)+13}-5) + \log(16)) - \\
& \quad \left. 3(16\log(2(-3t+\sqrt{6t(6t-7)+13}+1)) + 9\log(6t+2\sqrt{6t(6t-7)+13}-5)) \right) \\
& \frac{1}{12\sqrt{3}} \left(-108t^2 \log(-6t+2\sqrt{6t(6t-5)+7}+1) - 2\sqrt{27(t-1)t+7} - \sqrt{36t(3t-2)+13} + \right. \\
& \quad 6\sqrt{6t(6t-5)+7} - 12(1-3t)^2 \log \left(\frac{2(3t+\sqrt{6t(6t-5)+7}-2)}{\sqrt{36t(3t-2)+13}-1} \right) + 27(1-2t)^2 \log(2\sqrt{27(t-1)t+7}-1) + \\
& \quad \left. 108t \log(-6t+2\sqrt{6t(6t-5)+7}+1) - 27 \log(-6t+2\sqrt{6t(6t-5)+7}+1) \right) \\
& \frac{1}{12\sqrt{3}} \left(-\sqrt{108t^2+1} - 108t^2 \log \left(-\frac{6(3(1-6t)t+\sqrt{6t(6t-1)+1}-1)}{\sqrt{108t^2+1}-1} \right) - 2\sqrt{9t(3t-1)+1} + \right. \\
& \quad \left. 6\sqrt{6t(6t-1)+1} + 3(1-6t)^2 \log(2\sqrt{9t(3t-1)+1}-1) + 3(12t-1) \log(-6t+2\sqrt{6t(6t-1)+1}-1) \right)
\end{aligned}$$

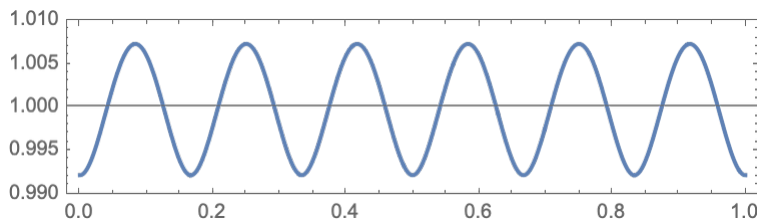
$6t = 1 \vee 3t = 1 \vee 2t = 1 \vee 3t = 2 \vee 6t = 5$
 $\frac{2}{3} < t < \frac{5}{6}$
 $\frac{1}{6} < t < \frac{1}{3}$
 $6t > 5$
 $\frac{1}{2} < t < \frac{2}{3}$
 $\frac{1}{3} < t < \frac{1}{2}$
 True

Integrating this over t we get the “average radius”, normalized to “circumference 1” from the fact that t varies from 0 to 1 going “around the circle”:

$$\frac{1}{54} (2\sqrt{3} + 3\sqrt{3} \log(3) + \log(2 - \sqrt{3})) \approx 0.145476$$

(This means that the “effective π ” for this circle is about 3.437.)

Now we can plot the “wobble” of the radius as a function of “angle” (i.e. t):



It looks a bit like a sine curve, but it's not one. And, for example, it isn't even symmetrical. Its maxima (which occur at odd multiples of 30°) are

$$\frac{12\sqrt{3} - 4\sqrt{7} - 3\log(3(7 - 4\sqrt{3})) + 6\log(\sqrt{7} - 2)}{48\sqrt{3}} \approx 0.146544$$

while its minima (at even multiples of 30°) are

$$\frac{1}{4\sqrt{3}} \approx 0.144338$$

and dividing by the average radius these are about 1.00734 and 0.992175.

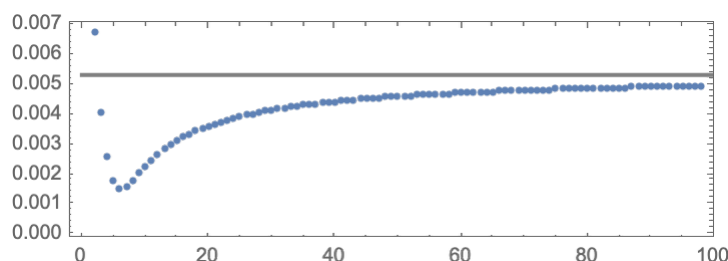
The ratio of maximum to minimum (effectively “wiggle amplitude”) is:

$$\sqrt{3} - \frac{\sqrt{7}}{3} + \frac{1}{4} \log\left(\frac{7}{3} + \frac{4}{\sqrt{3}}\right) + \frac{1}{2} \log(\sqrt{7} - 2) \approx 1.01529$$

Meanwhile, the standard deviation can be obtained as an integral over t , and the final result is

$$0.00533159$$

which is about 2.4 times larger than what we get at $n = 100$. We can see the approach to the asymptotic value by computing integrals over t for progressively larger numbers of discrete values of s (which, we should emphasize, is similar to values of n , but not quite the same, particularly for small n):



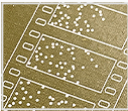
Posted in: [Historical Perspectives](#)

Join the discussion

Related Writings



Remembering the Improbable Life of Ed Fredkin (1934–2023) and His World of Ideas and Stories
August 22, 2023



A 50-Year Quest: My Personal Journey with the Second Law of Thermodynamics
February 2, 2023



How Did We Get Here? The Tangled History of the Second Law of Thermodynamics
January 31, 2023



The Making of *A New Kind of Science*
May 13, 2022

Popular Categories

- Artificial Intelligence
 - Big Picture
 - Companies and Business
 - Computational Science
 - Computational Thinking
 - Data Science
 - Education
 - Future Perspectives
 - Historical Perspectives
 - Language and Communication
 - Life and Times
 - Life Science
 - Mathematica
- Mathematics
 - New Kind of Science
 - New Technology
 - Personal Analytics
 - Philosophy
 - Physics
 - Ruliology
 - Software Design
 - Wolfram|Alpha
 - Wolfram|One
 - Wolfram Language
 - Other

Writings by Year

2023 | 2022 | 2021 | 2020 | 2019 | 2018 | 2017 | 2016 | 2015 | 2014 | 2013 | 2012 |
2011 | 2010 | 2009 | 2008 | 2007 | 2006 | 2004 | 2003 | All